

**MICROHOBBY**

# AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

*Especial*

AÑO I N.º 1

350 Ptas.

**SERIE ORO**

**CONTROL DEL IVA  
POR ORDENADOR**

**PROGRAMAS DE GESTION:**

**PONGA SU AMSTRAD  
A TRABAJAR**

**RSX:**

**NUEVOS COMANDOS  
PARA EL BASIC**

**QUE SE PUEDE HACER  
CON UN ORDENADOR EN CASA**

**LOS SECRETOS  
DE LOS FICHEROS  
ALEATORIOS EN DISCO**

**SOFTWARE**

**SUPERDIEZ:  
LOS MEJORES JUEGOS  
DEL MERCADO**

*H. Esco*

**HOBBY PRESS**



# AMPLIA LAS POSIBILIDADES DE TU AMSTRAD



**IMPRESORA PRINTER 130**  
Especialmente recomendada para ordenadores AMSTRAD. 94.900 Pts



**JOYSTICKS**  
Los famosos SVI de la serie Quickshot. Desde 1.600 Pts



**LAPIZ OPTICO**  
Diseña gráficos y menús de comunicación en la pantalla a color. Incluye software. 4.500 Pts



**INTERFACE SERIE RS 232 C**  
Para conectar con modems, impresoras serie u otros ordenadores. 11.750 Pts.



**UNIDAD DE DISCO**  
Incluye Sistema Operativo CP/M y lenguaje LOGO.  
(con controlador). 45.500 Pts  
(sin controlador). 39.500 Pts



**SINTETIZADOR DE VOZ**  
Emula la voz humana. Incluye dos altavoces y el software. 9.000 Pts



**MODULADOR TV COLOR**  
Para utilizar el TV como pantalla a todo color. 9.000 Pts (CPC 464), 9.450 Pts (CPC 664 y 6128)

**AMSTRAD** ESPAÑA

GRUPO INDESCOMP

Avda. del Mediterráneo, 9. Tels. 433 45 48 - 433 48 76. 28007 MADRID

Delegación Cataluña: Tarragona, 110 - Tel. 325 10 58. 08015 BARCELONA



**Director Editorial**

José I. Gómez-Centurión

**Director Ejecutivo**

Víctor Prieto

**Subdirector**

José Mario Díez

**Redactor Jefe**

Marta García

**Diseño gráfico**

José Flores

**Colaboradores**

Francisco Portolo, Pedro Sudón

Miguel Sepúlveda,

Francisco Martín,

Jesús Alonso, Pedro S. Pérez,

Amalio Gómez,

Juan J. Martínez,

David Sopuerta, Alberto Suárez,

Eduardo R. Velasco,

Javier Barcelo

**Secretaría Redacción**

Carmen Sotomayor

**Fotografía**

Carlos Condel

Javier Martínez

**Portado**

M. Barco

**Ilustradores**

Javier Iguol, J. Pons, F. L.

Frontán, J. Septien, Pejo, J. J.

Moro, Luigi Pérez, J. Siemens

**Edita**

HOBBY PRESS S.A.

**Presidente**

Mario Andino

**Consejero Delegado**

José I. Gómez-Centurión

**Jefe de Publicidad**

Concha Gutiérrez

**Publicidad Barcelona**

José Golón Cortes

Tel: (93) 303 10 22/313 71 62

**Secretaría de Dirección**

Moriso Cogorro

**Suscripciones**

M.º Roso González

M.º del Mar Colzodo

**Redacción, Administración y Publicidad**

La Gronjo, 39

Polígono Industrial de Alcobendas

Tel.: 654 32 11

Telex: 49 480 HOPR

**Dto. Circulación**

Carlos Peropadre

**Distribución**

Coedis, S. A. Valencia, 245

Barcelona

**Imprime**

Gráficos Reunidas

Avda. Aragón, 56 (MADRID)

**Fotocomposición**

Navacamp, S.A.

Nicolás Morales, 38-40

**Fotomecánica**

GROF

Ezequiel Solano, 16

**Depósito Legal:**

M-5836-1986

**Derechos exclusivos**

de la revista

**COMPUTING with the AMSTRAD**

Representante para Argentina, Chile, Uruguay y Paraguay, Cia. Americana de Ediciones, S.R.L. Sud América 1.532. Tel.: 21 24 64, 1209 BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace necesariamente solidaria de las opiniones vertidas por sus colaboradores en los artículos firmados. Reservados todos los derechos.

Se solicitará control OJD

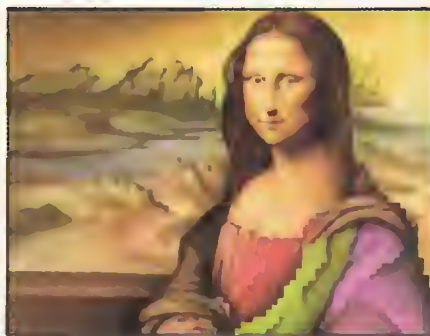
Año 1 • Número 1 • Marzo 1986  
Precio 350 ptas. Canarias, Ceuta y Melilla 335 ptas.

## 4 HISTORIA DE LA INFORMÁTICA

Los ordenadores han evolucionado muchísimo desde el ENIAC y el MARK hasta los máquinas de procesamiento paralelo basados en iconos y ventanos de nuestros días.

Pero resulta que la idea del ordenador se le ocurrió a la gente hace mucho tiempo, bastante antes del siglo XX, aunque, naturalmente, adoptó otros formas.

Detrás de esto hay todo un historio que merece lo peno ser contado.



## 8

### AMX MOUSE

Dibujar por ordenador disto de ser tarea sencilla si se hace «a pelo». Probablemente lo único formo racional de abordar el arte por ordenador es lo que preconiza AMX Mouse.

## 12 FICHEROS EN DISCO

De la importancia de este tema poco hoy que decir; todo el mundo sabe que se trata de algo crucial. Nosotros también lo sabemos, y por eso hemos escrito este artículo que explica completamente su funcionamiento.



### RELATO

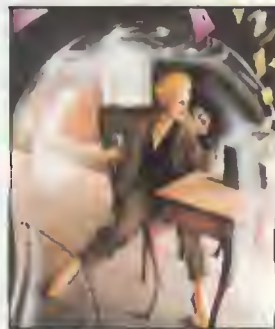
## 36

En un número especial, dedicado a algo como lo informático, siempre en lo punto de lo lonzo de lo tecnología, de coro al futuro, no podía faltar un pequeño relato que planteo uno escolofronte alternativo a nuestro porvenir

## 40

### COMO TRABAJA UN PROGRAMADOR

¿Quién hace los juegos? ¿Y cómo? Nadie mejor que un programador profesional para dar respuesta a todas estas interrogantes.



## 46

### LA INFORMÁTICA EN CASA

Se nos ocurrió pensar que sería útil hablar acerca de cómo y para qué puede usarse un ordenador en casa. Hasta qué punto es necesario e importante para nuestro vida y nuestros hijos, o no lo es.

## 56

### ESPECIAL LIBROS

Cuando un ordenador alcanza el éxito entre la gente, como en el caso del Amstrad, comienzan a aparecer en la calle ríos de tinta acerca de él.

Entre todo este farrago de libros, consumidos por el programador o usuario ávido de aprender e informarse, hoy de todo: bueno y no tan bueno.

AMSTRAD Especial ha seleccionado unos cuantos para someterlos al cuchillo del crítico.

## 60

### TODOS LOS RSX

En nuestro número especial los programadores y estudiosos del código máquina no van a quedar olvidados. Así que les ofrecemos un programa que contiene todos los comandos RSX publicados por AMSTRAD Semanal y unos cuantos nuevos especialmente creados para la ocasión, naturalmente en forma de cargadores Basic y listado desensamblado.

M. AMSTRAD ESPECIAL 3



# HISTORIA DE LA INFORMATICA

**La luz de la zona oeste de Filadelfia palideció tímidamente. En el primer piso de la Facultad de Ingeniería de Pennsylvania cientos de kilómetros de cable y más de 18.000 válvulas cobraban vida. Eniac comenzó a tragar tarjetas, la respuesta no se hizo esperar; en su interior salvo electrones, nada se había movido?**



El sueño de las hambrunas, la ilusión de varios siglos se había conseguido. Eran los finales de 1945.

## ...¡Y me llevo una!

La cosecha había sido buena, como cada año, el agricultor se dispuso a contar los sacos de trigo, por cada saco levantaba un dedo, pero yo no le quedaban dedos y aún sobraban sacos, costernado se miró las pies e intentó levantar el dedo meñique, lo olvidó y se lo impedía. Armándose de una seguridad que no le pertenecía gritó: **«y me llevo una»** al tiempo que para no aliviarlo levantaba su pie derecho. Bajó los dedos y proseguía contando normalmente. Por fortuna el número de sacos no pasó de los veinte y el inteligente agricultor no hubo de levantar su pie izquierdo. El sistema decimal ya existía.

Hicieran falta más de 2000 años para que este sistema se impusiera en Europa de forma definitiva. Después de su importación llevada a cabo por los árabes, el sistema de numeración indoárabe llegaría incluso a estar expresamente prohibido. Gracias a este sistema la automatización de los procesos de cálculo se haría posible.

Empezar un historio de la informática mil años antes de Cristo puede parecer ridículo, quizás nuestras decimales cabezas casi lo afirman. Sin embargo, no olvidemos que en la antigüedad cada cantidad se representaba por un símbolo, o un conjunto de éstos, en atención a un puñado de reglas más o menos cabalísticas, arcaicas y coópticas; el hecho de poder representar cualquier cantidad mediante

diez símbolos distintos y con unas reglas tan sencillas como fijar constituirían un primer paso hacia una sociedad informatizada.

El primer antecedente histórico destacable como forma de automatizar los cálculos data del 1100 o.c. el ábaco. Mediante este ingenioso instrumento constituido por varias cuentas ensartadas en unos varillos de alambre, se realizaban con notable presteza y fiabilidad, las cuatro operaciones básicas en la milenaria China. Para las más incrédulas, sólo decir que tras la invasión americana del Japón, en el 1946, un oficial norteamericano luchó en una competición aritmética contra un funcionario japonés. Como ormos se utilizaron uno calculador eléctrico de mesa por parte americana y un ábaco por parte japonesa. El resultado fue más que ridículo. **«Ayer la era de la máquina dio un paso atrás»**, fue uno de los titulares de la prensa norteamericana del día siguiente.

## Tic, tac seis por cuatro veinticuatro

Aunque la idea estaba en la mente de todos, el primero verdaderamente poseído por esta ambición fue Wilhelm Schickard, maestro relojero, quien en 1623 construyó su «reloj de cálculo». La máquina en cuestión realizaba las cuatro operaciones básicas y disponía de una capacidad de cálculo de seis dígitos. Curiosamente cuando mediante cualquier operación se superaba este límite (999.999) sonaba un campano, el operario entonces debía poner un anillo en su mano derecha para de este modo seguir efectuando las operaciones sin perder un millón en el cómputo.

Schickard escribiría a Kepler explicándole su invento, por el cual el eminente astrónomo se sentiría especialmente interesado. Sin embargo, el fuego acabaría con el reloj de Schickard le había construido. Kepler jamás llegaría a ver la máquina en funcionamiento.

Casi veinte años más tarde, Blaise Pascal, basándose en los engranajes de Schickard

construía su sumadora, la Pascalina. La Pascalina sólo era capaz de realizar sumas y restas. Como dato innovador respecto a la máquina de Schickard, fue lo resta realizado como suma de complementos. El sistema se basaba en la siguiente igualdad:

$$b-a = (b-(c-a))-c$$

Donde  $c$  era 10, 100, 1.000... en función del número de dígitos de  $b$ . Naturalmente, ningún ser humano emplearía la fórmula para calcular mentalmente, sin embargo, Pascal entendía que el objetivo no era emular el pensamiento humano para resolver los cálculos sino realizar éstos de la forma más rápida y sencilla y la Pascalina así lo hacía.

Ya podemos separar de la búsqueda de estas formas más o menos artificiosas de calcular, el nombre de John Napier. Inventar de los logaritmos. Mediante el empleo de sus famosas tablas saber multiplicar ya no era necesario, bastaba con saber que:

$$\log(a \cdot b) = \log(a) + \log(b)$$







Así, para multiplicar 3.457 por 768, bastaba con buscar sus logaritmos en la tabla correspondiente (8.1481564 y 6.6437897), sumar los datos obtenidos (14.7919461) y buscar su antilogaritmo, es decir, el número al que correspondió ese logaritmo (2654976).

Naturalmente obvia decir que el problema de la división queda igualmente resuelto mediante la aplicación de:

$$\log(a/b) = \log(a) - \log(b)$$

El descubrimiento de Napier engendraría el reglo de cálculo, instrumento desarrollado por el astrónomo Edmund Gunter alrededor del 1600 y perfeccionado por Seth Partridge en el 1662. Los cálculos, en ésta, se realizaban mediante mediciones de distancias geométricas sobre una escala logarítmica.

La diferencia ordenador digital/analógico comenzaba a vislumbrarse. En las calculadoras construidas a base de engranajes todos los números representados entre dos dientes, se

perdían, o bien avanzaba un diente o no avanzaba, no existían movimientos fraccionarios entre dos dientes. Por el contrario, en la regla de cálculo podía representarse cualquier número, bastaba con situar el puntero sobre la posición correcta de la escala. El problema estaba en conseguir por parte del operador y de la naturaleza intrínseca de la máquina, precisiones más allá del medio milímetro. El enfrentamiento entre estos dos conceptos antagónicos habría de resolverlo el futuro, razones económicas doblegarían el avance de la informática por una senda preferentemente digital.

Aunque sus estudios no fueron aplicados en su época, un hombre que contribuyó al posterior desarrollo de esto técnico, fue Gottfried Leibnitz quien desarrolló la aritmética binaria, orden en el que funciona cualquier computador de nuestros días. También realizaría algunos trabajos sobre lógica, en los que Boole se inspiraría años más tarde, para el desarrollo de la lógica simbólica. Sin embargo, Leibnitz no podía permanecer al margen de los vientos pragmáticos de la época, construyendo lo que él mismo denominó como calculadora universal. Esta aventajaba a la Pascalina al ser capaz de realizar las cuatro operaciones básicas. El primer modelo estuvo terminado en 1671, aunque no sería hasta 1694 cuando saca el modelo definitivo. El sistema de cálculo empleado revolucionó el ingeniería aplicado a estas máquinas. Incluyó un nuevo elemento que venía a sustituir la práctica totalidad de los engranajes. Estaba constituido por un cilindro dentado de longitud variable, cada uno de los dígitos se representaba a una altura distinta de la base del cilindro. El sistema, conocido como rueda escalonada de Leibnitz, continúa aún empleándose en muchos de los modelos de máquinas registradoras, que aún pueden verse sobre los mostradores de pequeños comercios.

La calculadora de Leibnitz fue la primera en comercializarse a razón de unas 20 ó 30 por año. Por el momento la industria era incapaz de realizar en serie piezas que debían soportar tal grado de precisión. Aún habría de esperar varios años para conseguir un volumen de fabricación lo suficientemente importante para reducir los costos de estas pequeñas maravillas artesanales.

### Una historia frustrada

A principios del siglo XX Chales Babbage se plantea la empresa de construir una máquina que calcule logaritmos con una precisión de veinte decimales, los tablos confeccionados por Napier al final de su vida estaban sembrados de errores, de forma que los resultados obtenidos a partir de mediciones, realizados con suma pulcritud, resultaban falsos al haberse utilizado en su cálculo valores erróneos de los logaritmos. Barcos que se salían de su ruta, paralelos y meridianos más largos

o cortos de lo que en realidad eran, constituían si bien no una norma sí un hecho manifiesto de estos errores.

Con una subvención de más de 17.000 libras (unos 200 millones de pesetas) Babbage acometió esta empresa por encargo de la Royal Society. La mente inquieta y asombrosa del científico perfeccionaba cada noche el proyecto del día anterior, la máquina de diferencias, nombre con el que bautizó su proyecto, en función del sistema de cálculo que utilizaba, basado en la aproximación de diferencias polinómicas. Los mecánicos que trabajaban con él en el proyecto y que no entendían aquellas modificaciones encaminadas a una optimización del sistema, acabaron por enfrentarse con su persona y lo subvención que le otorgara la Royal le sería finalmente retirada.

La decisión de la Royal no desanimó a Babbage, todo lo contrario, abandonó el proyecto o mejor dicho lo cambió por otro mucho más ambicioso. La máquina analítica.

La concepción ideal de esta máquina era prácticamente idéntica a lo que hoy concebimos. Un dispositivo de entrada y otro de salida, una unidad de control, una memoria para almacenamiento de los datos y una unidad lógica capaz de realizar ciertas operaciones, constituían a grandes rasgos las diferentes partes de su máquina.

Especialmente revolucionario fue el empleo de tarjetas perforadas como medio de introducir los datos en la máquina. Realmente el sistema no era propiamente suyo, aunque sí su utilización. El inventor del sistema fue Jacquard el cual las empleaba en la confección de tapices. Babbage llegaría a afirmar: «**mi máquina tejerá pautas algebraicas de la misma manera que el telar de Jacquard teje flores y hojas.**»

Sin embargo, poco sería lo que la máquina de Babbage «tejería», la técnica habría de traicionarle. Frustrado ante la imposibilidad práctica de la construcción de su máquina, moriría amargado llevando consigo la práctica totalidad de sus ideas. Lo poco que de él nos queda se lo debemos a Lady Lovelace, hija de Lord Byron y discípula suya, su verdadero nombre Augusta Ada, en cuyo honor la NASA bautizaría con su apellido el lenguaje por ellos desarrollado.

Un concepto básico diferenciaba el prototipo de Babbage de las máquinas calculadoras diseñadas hasta la época. La máquina analítica era programable, llegándose incluso a considerarse en ella los saltos y bifurcaciones durante la ejecución de un programa. Las bases teóricas estaban creadas, no en vano Aiken, creador del Mark I, afirmaría un siglo más tarde: «**Si Babbage hubiera vivido 75 años más tarde yo estaría sin trabajo.**»

Ya que nos encontramos en el siglo XIX no podemos olvidar, y no es chauvinismo, un nombre español en esta historia, Leonardo Torres Quevedo, desarrolla dos máquinas calculadoras, una basada en conceptos analó-



gicos y la atra en digitales. Sin embargo, su verdadero triunfo en este campo la constituyó su Ajedrezista, máquina totalmente automática capaz de dar mate con rey torre contra rey. La perfección de esta maravilla alcanzaba a detectar jugadas no válidas realizadas por el oponente.

### La historia de un censo

La incorporación de motores eléctricos a estas máquinas consiguió acelerar notablemente las procesas de cálculo así como su fiabilidad, aunque la concepción de calculador concebida por Leibniz no sufriría cambios sustanciales. En 1886 los horizontes de lo informático habrían de verse nuevamente ampliados.

Según la ley americana, cada diez años debía realizarse un censo de población, en el correspondiente a la etapa comprendida entre 1870 y 1880 se habían empleado 8 años y esta cifra aumentaba geométricamente a medida que aumentaba la población. Hermann Hallerith, funcionario de la oficina del censo, se planteó el problema de automatizar la clasificación. Para su objetivo recabó la idea de Babbage y codificó mediante perforaciones cada una de las características que el censo recogió. El sistema de lectura de estas fichas era relativamente sencilla. La tarjeta se situaba de forma automática entre las planchas sobre las cuales se encontraban distribuidos unos contactos eléctricos, allí donde pudiese encontrarse una perforación. De este modo la máquina era capaz de registrar los resultados del individuo según circulase o no la corriente eléctrica; si existía perforación los contactos se tocaban, circulando la corriente eléctrica, en caso contrario esto no ocurriría.

La máquina de Hallerith fue todo un éxito logrando reducir el tiempo de clasificación de 8 a 2 años, unas trescientas tarjetas por minuto. Conceder del potencial de su invento, Hallerith patentó su invento fundando en torno suya una empresa, la TMC (*Tabulating Machine Company*).

La principal innovación de la clasificadora de Hallerith consistía en procesar información exclusivamente numérica como era el caso de los calculadoras anteriores, sin cualquier dato capaz de expresarse mediante una combinación de «síes» y «noes».

Años más tarde la TMC pasaría a formar parte del Holding TCR cuyo director general sería a partir de 1910 Tamos Watsan, el cual no tardaría mucho tiempo en hacerse dueño de la compañía, el cual cambiaría su nombre por el de IBM (*International Business Machines*). El verdadero boom de esta empresa sobrevino al finalizar la gran crisis del 29, mientras que la industria había quedado paralizada, IBM había seguido trabajando a pleno rendimiento como si nada sucediese. El stock almacenado había llegado a alcanzar cotas alarmantes. Terminada la crisis que comenzó un viernes en Nueva York, la implantación de

la Seguridad Social en las EEUU provocaría una demanda brutal de todos los productos que comercializaba esta firma. Las empresas en atención a unos mayores rendimientos buscaron en la mecanización de la pieza clave. Sólo gracias a la labor de hormiguita hacendosa, realizado durante los últimos años, IBM sería capaz de enfrentarse a la avalancha de pedidos. Como dato significativa, sólo resaltar las tres mil millones de tarjetas vendidas en 1935 y que pueden dar una idea de todo lo dicho.

### ¡Eureka! ¡programables!

A partir de 1934 Konrad Zuse desarrolla en Alemania el Z1 máquina destinado a la resolución de problemas estadísticos, sin embargo el nozismo no sería el caldo de cultivo ideal para el desarrollo de su invento. El Z1 se basaba en su totalidad en la aritmética binaria, operaba en cama flatante y disponía de 64 registros de 22 bits cada uno. Al Z1 le siguió el Z2 y a éste el Z3 el cual fue considerado como el primer ordenador universal y completo. Antes del inicio de la II Guerra Mundial Zuse propuso al gobierno nazi la creación de un ordenador de características similares al Z3 pero electrónico, la magnitud del proyecto no fue comprendido en su verdadera dimensión siendo rechazada. La gloria aguardaría unas años más hasta el nacimiento del Eniac.

Comenzada la guerra, toda la serie Z de Zuse constituyó una de las principales blancas de la aviación aliada, sólo el Z4, versión modificada y mejorada de su antecesor el Z3, se salvaría en esta caza.

Paralelamente a la serie Z, la ATT en Estados Unidos no escatimaba esfuerzos en esta industria floreciente que habría de convertirse en Ciencia. Con una subvención de esta compañía telefónica, George Stibitz finalizaría en 1940 su Complex Calculator, ordenador electromecánico que al igual que el de Zuse, se basaba en ceros y unos. Como aportación revolucionaria, Stibitz incorporó a su máquina un teletipo que se encontraba conectado, vía telefónica con otro situado en Nueva York a más de 400 kilómetros de distancia. La conversación telefónica entre los ordenadores era una hecha.

El pesimismo de una guerra que embarcaría a toda Europa constituyó indudablemente un impulso considerable en el desarrollo de lo informático. Los cálculos balísticos y los guerreros de mensajes cifrados conformarían los dos objetivos básicos de los máquinas pensantes en la primera mitad del la década.

En 1939, con una subvención de IBM, Howard Aiken comienza a trabajar en el que sería conocido oficialmente como ASCC, Automatic Sequence Control Computer, conocido cariñosamente como Mark I. Con un precio superior a cinco millones de dólares y más de 800 km de cable en su interior, Mark reolaba sumas y restas en un tercio de segundo,

dividiendo en tan sólo 10 segundos. El dispositivo de entrada se basaba en cinta continua perforada y naturalmente era programable. Asimismo disponía de un conjunto de mecanismos que le permitían calcular las funciones más usuales, seno, coseno, tangente, etc...

### El primer coloso

Casi de forma paralela a la construcción de Mark, en la Facultad de Ingeniería Moore de Pensylvania, comenzaron a desarrollarse el primer ordenador electrónico. Existía ya un precedente en la aplicación de la técnica de válvulas para estas máquinas. Jhon Atanaroff, físico norteamericano, la había empleado en la construcción de un calculador binario electrónico. En 1972 un juez estadounidense habría de conceder, mediante sentencia, la gloria oficial de haber sido el inventor del ordenador electrónico. Si bien no podemos quitar el mérito a este hombre de ciencia, no sería por menos que injusto otorgar la sentencia del caso Atanaroff. La calculadora, diabólicamente rápida coreció de la cualidad indispensable del concepto de ordenador, hoy ya bien definido, coreció de la posibilidad de ser programado.

Dos hombres serían los encargados de acometer la magna empresa, Hohn W. Muchly y Jhon Eckert. De esta unión nacería Eniac, el primer ordenador electrónica del mundo. Con más de 18.000 válvulas y una producción de calor equivalente a 200 kw, mientras estaba encendido, Eniac realizaba una suma en 0,6 milisegundos y en una división tardaba menos de 10 milisegundos. Sin embargo, cada día de trabajo de Eniac suponía un numeroso grupo de válvulas fundidas lo que provocaba un continuo paseo de operarias por toda la primera planta del edificio, o la búsqueda de las bombas fundidas. A parte de esto la programabilidad de Eniac era aún más que abasaleta. Cuando se precisaba cambiar de programa la estructura interna de Eniac había de ser modificado en función de las nuevas necesidades.

### Programas y datos, sólo uno

En 1945 Johannes van Neumann habría de subirse al tandem, Eckert/Mouchly. De esta unión habría de nacer en 1952 el EDVAC con un programa almacenado en una memoria de 8 K.

La máquina de Neumann se basaba en tres principios básicos, que se conservarían hasta nuestros días:

I) El programa debe ser almacenado en la misma forma que los datos.

II) Debe existir una instrucción lógica condicional que permita dotar al ordenador de capacidad lógica.

III) El programa debe constituir una cadena de decisiones lógicas binarias.

Estos tres principios básicos provocarían la





aparición de una necesidad o un no sospechado, los lenguajes de programación.

El recién creado departamento de software creado por IBM y cuyo cargo se encontraba John Backus sería el encargado de desarrollar el Speed-coding para el IBM 701, un ordenador con una aplicación puramente científica.

### La fabricación en serie

Lo primero firma comercial que acometió con éxito esta tarea fue la Remington Rand, pero la cual trabajaban Eckert y Mouchly, quienes después de abandonar la Facultad de Moore por problemas de patente sobre la Eniac, habían sido rechazados por Watson para su incorporación en IBM. De este modo en 1951 surgió el UNIVAC 1, potente ordenador administrativo. A este computador se le incorporaron diversos avances, pero el más revolucionario, lo constituyó sin duda el empleo de cintas magnéticas como soporte de almacenamiento. El rotundo éxito del Univac acabó por hundir la Remington. La empresa, conocedora de la alta competitividad de su aparato, olvidó la investigación dedicándose de forma casi exclusiva a la comercialización de su maravilla. Cuando la IBM anunció la aparición de su modelo 705 lo sentenció contra la Remington ya estaba escrito. Lo respuesta de la Rand fue el UNIVAC 2, pero, ya era tarde. Producto de este fructífero negocio la Sperry Rand como fusión de Sperry Gyroscope y lo ya mencionada compañía. El golpe final de IBM fue el 1401 que en 1959 llegó a opoderarse totalmente del mercado.

El modelo 704 de IBM era copo de trabajo en Fortran y poro él un grupo de usuarios desarrolló un sistema operativo, el cual, sería distribuido por IBM, de forma totalmente gratuita.

### Los primeros hijos

El descubrimiento, por parte de Bardeen y Britton, del transistor y su posterior perfeccionamiento por Shockley, en 1951, convulsionaría nuevamente la concepción técnica, que no lógico, del ordenador. Los módulos se hacen mucho más pequeños, menos caros y enorme-

mente más rápidos. La memoria constituida en su mayor parte por bobinas magnéticas en los ordenadores de la primera generación eran sustituidos por núcleos de ferritas. Asimismo la comunicación por medio de fichas, se hizo mucho más rápida y como dispositivo de almacenamiento comenzaron a utilizarse impresoras.

Todo esto en lo que respecta al Hardware o físico del ordenador, el software o lógico debería también sufrir un notable impulso con la aparición de lenguajes mucho más desarrollados como el Algol, Cobol y Lisp. Del primero de ellos podemos destacar su formalidad y estructura, el Cobol supondría la panacea en el manejo de ficheros y el Lisp, un lenguaje creado durante la 2.ª generación y que no ofrecerá su verdadero juego hasta la llegada de la ondiadamente esperada quinta.

Los primeros firmes en utilizar la nueva tecnología fueron RCA y NCR a finales de los cincuenta. IBM lanzó también varios modelos el 7070, el 7080 el 1620, y el más pequeño de todos, el 1401 cuyos ventas superaron las 20.000 unidades. Astronómica cifra si consideramos los precios que alcanzaban estos equipos.

### Ayer la tercera generación

Lo base sobre lo que se opesentó la tercera generación fue el circuito impreso. La velocidad de proceso había que medirlo ya en millonésimos de segundo. Junto a esto innovación surgió la idea de la multiprogramación, gracias a esto varios usuarios compartían, desde distintos terminales, una misma unidad central. Realmente podía llegar a pensarse que aquellos pequeños terminales constituían por ellos mismos un ordenador. Se desarrollaron varios lenguajes, entre ellos nuestro muy querido y apreciado BASIC. Inicialmente este lenguaje, que más tarde se convertiría en el más popular, fue concebido por Kemeny y Kurtz como medio de introducir a los estudiantes en el conocimiento y manejo del FORTRAN. Fue igualmente desarrollado el PASCAL diseñado a partir del ALGOL 62. Como producto del matrimonio entre el Fortran y el Cobol nació de las monas de IBM el PL/I.

La tecnología se había convertido en un coral desbordado y lo informático no había de soltarse de uno de sus bridos. La aparición de la MSI (*integración a media escala*) provocó la reducción de los monstruos concretándose en la aparición de los miniordenadores en los inicios de los 70.

La MSU vendió o desplazó a la MSI. Mediante integración a gran escala se conseguirían los primeros microprocesadores. El primero de éstos en comercializarse, sería el I-4004 por parte de Intel, con 16 registros de 4 bits y un acumulador, realizaba sumas de 4 bits a velocidades increíbles. Al I-4004 le siguió el I-8008, ya de ocho bits, el M-68000 y en 1976 apareció el popular Z80 con más de 8.000 transistores en su interior.

## Hoy

La utilización de los microprocesadores haría posible el acceso de cualquiera a un microordenador. El primero en comercializarse al público fue el ALTAIR 8000, con un precio de unos 600 dólares no tenía competencia, en el aspecto económico claro está, con ningún otro equipo, el precio inmediatamente superior más bajo rondaba los 600.000 dólares. Después fue el 680 con un procesador de 8 bits, el Motorola 6800. Paradójicamente las compañías pioneras en dar el gran salto hacia un público más general fueron desplazadas por el empuje de los más fuertes.

Especialmente revolucionario fue la aparición de la saga PC de IBM, hasta tal punto que hoy día decir compatible es equivalente a decir compatible PC. APPLE fue otra de las compañías que entraron con pie firme en este nuevo acontecimiento, con su APPLE I puesto a punto por Steve Jobs y Steve Wozniak. Últimamente discrepancias de éste con la compañía le han hecho marcharse, quizás quien sobre, hoyo vuelto a su gorroje.

Con la aparición del ordenador doméstico lo que fue la base de las novelas de ciencia ficción de los años 70, habría de convertirse en realidad; un hogar, un ordenador. Son dignos de mención en este hecho compañías que hoy o todos nos suenan: Sinclair, Atari, Commodore, todas las acogidas a lo como MSX, Amstrad...

## Mañana

Lo que el futuro nos depore nadie puede predecirlo. Se ha llegado a afirmar que si la aeronáutica hubiera avanzado en la misma medida en que lo ha hecho lo informático, durante los últimos 25 años, un Boeing 767 costaría unos 60.000 pesetas, daría la vuelta al mundo en 20 minutos y el costo de combustible no alcanzaría los 20 litros. Con este posodo increíble quién puede imaginar un futuro.

Lo coreado quinto generación, lo último opuesto japonés, puede volver a cambiar el rumbo de lo informático, como ya lo hicieron las ideas de Boboche. Todo como materia reservada, de sus contenidos sólo se conocen sus fines, lograr que los ordenadores comprendan el lenguaje humano. Muy probablemente los compiladores de Pascal, Cobol o Fortran hoy que cambiarlos por intérpretes de inglés, francés o euskero. La aplicación del procesamiento en paralelo la velocidad de los ordenadores, hasta ahora medido en MIPS (*millones de instrucciones por segundo*).

El producto de todo esto quizás sea lo más parecido al HAL de la novela de Arthur C. Clarke «2001, una odisea en el espacio». Por cierto, conocía la ironía del nombre elegido, ¿no?, corra los letreros de sus iniciales una unidad en orden olfobético.

Por J.J. Martín



# AMX MOUSE

**Para hacer nuestro pequeño ordenador cada vez más parecido a los grandes ordenadores se crean constantemente utilidades de software y hardware, que le hacen, día a día, más potente y fácil de usar.**

Pedro S. Pérez



**AMX MOUSE**

SE es una de estas utilidades. La traducción literaria de este aparato no nos sitúa a la altura de su significado, ya que, como todo el mundo sabe, un mouse es un ratón, y este aparato no tiene nada que ver con los ratones, sino con una especie de joystick especialmente diseñado para la creación de pantallas.

Para empezar vamos a comentar el contenido de este paquete así como su utilidad.

## Contenido del paquete

Contiene una cinta o un disco con cuatro programas entre los que se hayan, un generador de pantallas, un generador de iconos, un generador de formas y un programa especial para inicializar el ordenador.

También dentro del paquete encontraremos un pequeño aparatito que deberemos conectar en la salida del port de joystick y a la vez a la corriente 5V, así como el mouse que lo conectaremos a ese interface.

Una vez conectado el aparato procederemos a la carga del programa AMX que inicializará el ordenador. Después realizaremos la carga del programa ART que es un programa con el que es posible la creación de pantallas y que sólo podremos hacer funcionar con el ratón.

Al terminar la carga de dicho programa, nos encontraremos con una pantalla dividida en cuatro ventanas: en la de la derecha se encuentra la información de los iconos que nos permitirán la realización del dibujo,

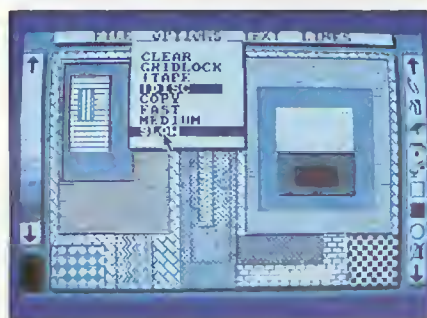
así como la herramienta que estamos utilizando, la información sobre el trazo de la brocha, el spray y el de la goma de borrar. La ventana de la izquierda contiene el dibujo que estamos utilizando para el fill, la brocha y el spray. Y por último en la superior encontraremos cuatro opciones que contienen el manejo de cinta o disco, el modo texto, modo de trazo y opciones diversas.

El Mouse (que a partir de ahora llamaremos ratón) se utiliza de la siguiente manera: al cargar el programa ART aparecerá en pantalla una flecha; al mover el Ratón la flecha se desplaza a través de la ventana llevando la misma trayectoria que nuestra mano. El botón de la izquierda se utiliza para ejecutar, el del centro para mover y el de la izquierda para anular.

## Los iconos de pantalla

En la ventana de la izquierda encontraremos unos dibujos que simulan aparatos de dibujo, esto es lo que se denomina un icono.

Empezando por la parte de arriba encontramos un lápiz; este icono se utiliza para representar la función de tiralíneas y su manejo es muy sencillo: con la ayuda del ratón y pulsando la tecla central (mover), desplazaremos la flecha hacia la posición en la que deseamos trazar una línea; una vez posicionada debemos pulsar el botón de la izquierda (fijar)



y volveremos a desplazarnos a la posición hasta donde vamos a trazar la línea; una vez en ella pulsaremos la tecla de la izquierda y ya tendremos la línea acabada.

El dibujo que representa una pistola de pintar es el icono que se utiliza para dibujar con distintos tipos de trazos. Así, si deseamos realizar una línea muy ancha, en lugar de utilizar el icono del lápiz utilizamos la pistola y, eligiendo un trazo ancho, lograremos realizar la línea de una sola vez. Otra de las ventajas del uso de la pistola es la de poder dibujar con un dibujo dentro del trazo, o sea, podemos dibujar la línea gruesa y a la vez que esta aparezca en pantalla con un dibujo en su interior de rayas.

Si al realizar un dibujo nos hemos deslizado hacia un lugar que no queríamos pintar, bastará con recoger el icono destinado a borrar, y que no es otro que un simil a una goma de borrar. Una vez recogido pulsaremos el botón de ejecutar para realizar el borrado del tamaño del trazo escogido.

El icono con forma de rodillo de pintor se utiliza de manera similar a la de la pistola pero no se puede fijar un trazo en pantalla a no ser que esté moviéndose el ratón.

## Resulta fácil rellenar y crear figuras...

Con imaginación podremos observar después una especie de copa y una jarra vertiendo líquido en ella;



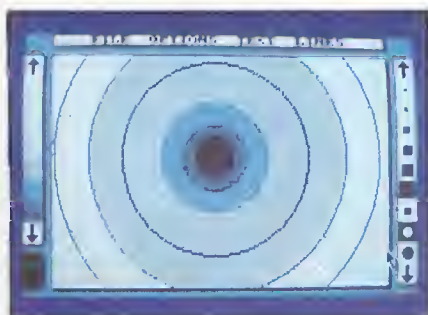


este icono representa la función fill (rellenado de figuras); con esta opción podemos rellenar cualquier superficie de una figura con el contenido que se encuentra en la parte inferior izquierda de la pantalla.

Si utilizamos el icono que representa un cuadrado, podemos realizar cualquier tipo de figuras de cuatro lados del tamaño que sea necesario. Para el uso de este icono en lugar del símbolo de la flecha aparecerá una cruz con un pequeño círculo en el centro. Si deseamos realizar un cuadrado tenemos que posicionar la cruz en el lugar elegido para uno de los vértices del cuadrado, y pulsar la tecla izquierda del Ratón. Después nos desplazaremos hasta el lugar elegido y opuesto a ese vértice y pulsaremos la tecla central.

Debajo de la figura cuadrada se encuentra otro cuadrado: su utilización es idéntica al icono anterior, pero al terminar de realizar el cuadrado éste se rellena automáticamente de el color (blanco, negro o invertido) que tengamos definido.

Para hacer un círculo el programa posee un icono con forma redonda.

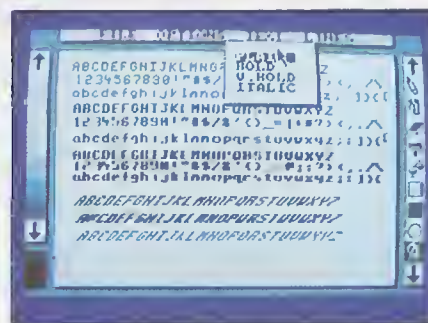
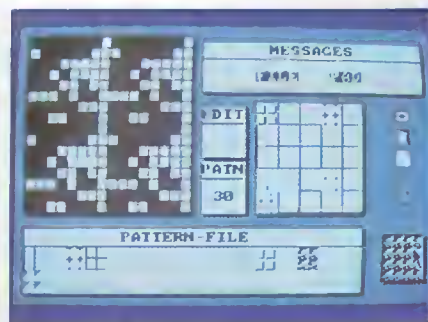


Para utilizar esta opción debemos colocar la cruz en el lugar que deseamos sea el centro del círculo y pulsaremos la tecla de la izquierda; después nos desplazaremos hasta el límite que deseamos tenga el círculo y pulsaremos la tecla central. En la pantalla aparecerán los trazos de un círculo pero sin completar; si no hemos obtenido el resultado esperado pulsaremos la tecla de la derecha y el trazo desaparecerá, por el contrario, si el trazo nos es satisfactorio, pulsando la tecla de la izquierda el círculo se completará.

### ...así como distintos tipos de letra

El icono que corresponde a la letra A es el utilizado para el modo de texto, con esta función podemos escribir en pantalla la letra que deseamos y en cualquier posición de la pantalla. Al elegir la opción de la Letra A la flecha se transformará en una especie de «I» mayúscula. El motivo de aparecer esta figura no es otro que la de facilitar la posición en la que deseamos introducir el texto sirviendo como guía la parte de la derecha de la I. Si deseamos escribir en cualquiera de los seis tipos distintos de caracteres que podemos utilizar, tendremos que guiar el cursor hacia la ventana superior y colocarlo encima de la palabra TEXT: al colocarnos encima de dicha palabra aparecerá otra ventana que a su vez contiene un pequeño menú de opciones entre las que podemos elegir dos tipos de letras con tres formas distintas de grosor de letra. Al pasar por encima de estas opciones, las letras se invertirán de color, pasando a ser el papel negro y la tinta blanca. Si deseamos utilizar alguna de ellas basta pulsar el botón izquierdo para fijar esa opción. Con ayuda del Ratón colocaremos el cursor en la posición donde vamos a colocar algún carácter, pulsaremos la tecla izquierda y, tecleando en el ordenador, los caracteres van a aparecer en pantalla. Una vez terminado el texto para volver al modo cursor pulsaremos la tecla central.

Colocando el ratón sobre las flechas que se encuentran en la ventana izquierda y pulsando el botón de ejecutar, los iconos se desplazarán hacia arriba y abajo apareciendo otros distintos y que se utilizan para otras funciones. Estos iconos nos indican el trazo que estamos realizando



do en ese momento así como todos los que podemos utilizar. Para comprender más fácilmente el significado que queremos definir con la palabra trazo podemos compararlo con dos rotuladores, uno gordo y otro fino; según el trazo que elijamos así será la línea gorda o fina con forma cuadrada o redonda.

En la ventana de la izquierda se encuentran una especie de tramas que son las que podemos utilizar para la opción de fill, pistola y rodillo. Para entenderlo más claramente diremos que es como dibujar de nuevo dentro del trazo un dibujo o una forma concreta; así, si deseamos llenar de rayas verticales una figura de cualquier tipo, colocaremos dentro del cuadro inferior de la izquierda el contenido que deseamos aparezca dentro de la figura y luego, con la opción fill, rellenaremos ésta de dicha forma.

### Manejo de ficheros desde AMX

En la ventana superior encontraremos las cuatro palabras que definen las siguientes opciones:

La opción FILE es la que contiene la posibilidad de carga de una pantalla, el salvado de ésta, la carga de otros tipos de fill y la función PRINT que utilizaremos para la impresora.

En el apartado OPTIONS se encuentra entre otras la opción de borrado de pantalla, el salto de la línea, la opción cinta o disco y la opción copy.



# Amx Mouse

SOFTWARE	ART	Programa especialmente indicado para el diseño de pantallas y su posterior utilización, a través de impresora. Util para diseñar modelos de impresos, tipa facturas, albaranes, membretes, etc.
	ICONDES	Con él podremos reformar o cambiar los iconos que utiliza el programa ART para una mayor facilidad de manejo por nuestra parte.
	PATTERN	Programa generador de formas a utilizar posteriormente con la función Fill.
HARDWARE	INTERFACE	Aparato especialmente diseñado para conectar el ratón al ordenador.
	RATON	Elemento de trabajo muy útil a la hora de dibujar. Movida gracias a una bola que se encuentra en su interior permite realizar dibujos trasladándolo sobre una mesa.

APARATO AMX	TIRA- LINEAS SI	PISTOLAS SI	BORRAR SI	BROCHA SI	• FILL SI	DEF.FILL SI
CIRCULOS SI	ELIPSE NO	CUADRADO SI	TEXTO SI	JUEGOS CARACT. 6	SIMETRIA NO	
IMPRESORA SI	TRAZOS 8	RALLADO PANTALLA NO	COLOR 2	DEFINIR TRAZO SI	SENSIBILIDAD 3 FORMAS	

Cuando empezemos a manejar el ratón notaremos que es excesivamente sensible su manejo y que nos cuesta llevarlo done nosotros queremos, para ello tiene en este apartado tres opciones que nos permiten ajustar la sensibilidad del ratón desde rápido a lento.

Dentro de opción TEXT se encuentran los distintos tipos de caracteres disponibles.

Y por último en el apartado LINES se encuentran el color que se utiliza. Disponemos de tres opciones dentro de esta opción: BLACK, INVERT y WHITE (Negro, Invertido y Blanco); eligiendo cada una de esas

opciones realizaremos el dibujo en blanco y negro. La forma de trabajo invertido se utiliza para realizar el dibujo de forma que se invierta en color o, lo que es lo mismo, pasando lo que es negro a blanco y viceversa.

## Y el resto de los programas

Para terminar comentar los otros programas que contienen este paquete.

El programa PATTERNS sirve pa-

ra crear nuevas formas a utilizar con el programa ART.

Con ayuda del ratón guiaremos la flecha sobre la ventana cuadrícula: pulsando el botón de la izquierda pondremos el cuadro en negro, pero nos fijaremos que a la vez aparecen cuatro cuadros más en la ventana. El motivo de estos cuatro cuadros es la de realizar una mayor simetría entre las formas con sólo realizar una figura de ocho por ocho pixels.

Una vez realizadas, las guardaremos dentro de la ventana con el nombre de PATTERN-FILE. Para trasladarlas a esta ventana pondremos la flecha entre las cuadrículas de la ventana de creación. Seguidamente aparecerá en el lugar de la flecha la forma realizada, y la trasladaremos hacia la ventana pequeña cuadrícula que se encuentra en el centro, que podríamos llamar zona de tránsito.

Cuando hayamos realizado todas las formas llevaremos la flecha hacia el icono que representa unanidad de disco y pulsando la tecla de la izquierda tendremos la opción de fichero.

Si por el contrario deseamos desear algunas de las formas realizadas, nos dirigiremos a el icono que asemeja una papelera y, si lo que deseamos es limpiar la pantalla, buscaremos el icono de hoja en blanco.

El último icono de esta opción es el que se utiliza para salir del programa.

Con el programa ICONDES podremos diseñar los iconos que deseamos utilizar en el programa ART.

La forma de utilización de este programa es similar a la del programa ART, moviendo el ratón sobre la ventana rallada realizaremos el dibujo del icono.

Después lo desplazaremos a la zona de salvado de iconos y los guardaremos para la posterior utilización con el programa ART.





# 3~D VOICE CHESS

**Ajedrez tridimensional con voz en castellano**

**Amstrad CPC 464, CPC 664 y CPC 6128**

**P.V.P.**

**2.300.- (cinta)**

**3.300.- (disco)**



PUBLISHED:

**DEEP THOUGHT  
SOFTWARE**

DISTRIBUTED:

**cd software**



Brillantes gráficos 3-D  
(calidad de color)

4 niveles de juego  
(de principiante a experto)

Rápidas respuestas desde 5 segundos

Producido en exclusiva para España por:

**ACE**

Actividades Comerciales Electrónicas, S.A.

Tarragona, 110 · Tel. 325 10 58 · 08015 Barcelona. Télex 93133 AC EE E

YA DISPONIBLE EN



... Y EN TODAS LAS  
TIENDAS ESPECIALIZADAS



# TRATAMIENTO DE FICHEROS EN DISCO

**Una de las mayores ventajas, por no decir la mayor, del uso de unidades de disco, es la velocidad de acceso a los datos almacenados en él. Esto se pone de manifiesto, sobre todo, en el tratamiento de ficheros.**

Javier Barceló



Hay tres maneras de acceder a un fichero: secuencialmente, por índices y directamente. De ahí el nombre de ficheros secuenciales, indexados y aleatorios o de acceso directo. De estos tres, los ficheros indexados son los únicos que no podemos utilizar en el AMSTRAD por no estar preparados para ellos. En el **AMSTRAD CPC 464** sin unidad de disco, sólo se pueden utilizar ficheros secuenciales, mientras que en el CPC 664 y CPC 6128 así como en el CPC 464 con unidad de disco, no tenemos por qué limitarnos a usar el acceso secuencial y podemos optar por el acceso directo. Pero **¡CUIDADO!** No nos engañemos. No siempre éste es más útil que aquél. Veremos que un programa que gestione ficheros de acceso directo ofrece más posibilidades, pero a cambio resulta más complejo que si maneja ficheros secuenciales. Seamos pues prácticos y veamos las posibilidades de cada uno, las similitudes y las diferencias entre ambos, y elijamos el que mejor se ocople a nuestras necesidades.

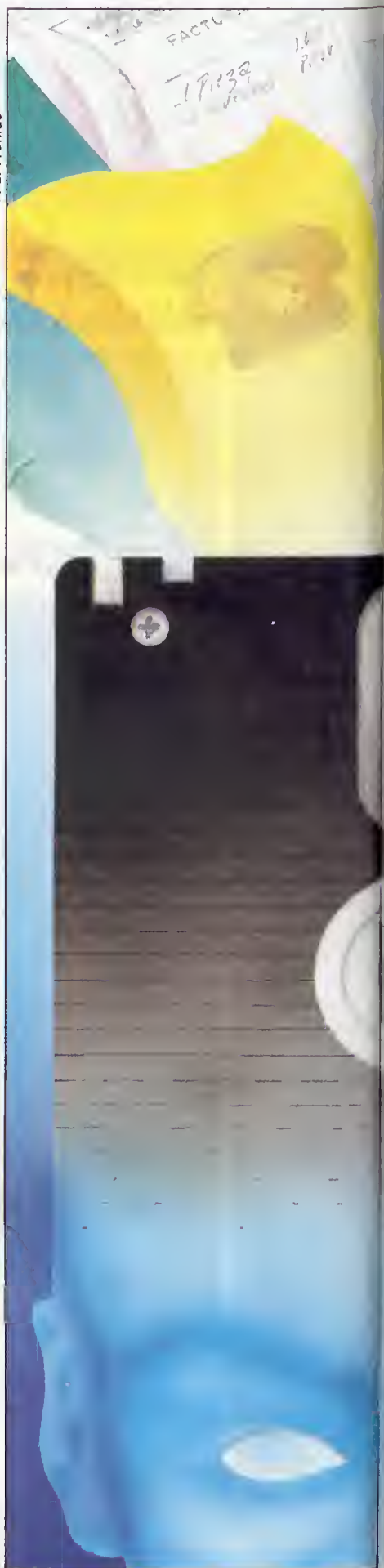
## Ficheros, registros y campos

A lo largo del artículo, estos tres palabras van a salir muchas veces, por lo que es esencial conocer bien su significado. Al hacer un programa, se nos puede plantear la necesidad de almacenar una serie de datos, de manera más o menos permanente para poder consultarlos o realizar otras operaciones posteriores.

Si los datos son pocos, y no vamos a haber necesidad de aumentarlos ni de modificarlos, se pueden poner en sentencia DATA, pero si por el contrario son muchos los datos o son datos que hoy que modifico con cierta frecuencia, habrá que crear un FICHERO. Esto significa crear un archivo, independiente del programa, donde los tenemos almacenados, y el programa es una herramienta que nos permite modificarlos, consultarlos, listarlos o cualquier otro cosa que necesitemos. A cada ficha que utilicemos lo llamamos REGISTRO. Un registro telefónico, con nombre, apellidos, dirección y teléfono, es un fichero, y el conjunto de todos los datos de una persona en este fichero compone un registro del fichero. Un registro no es más que un conjunto de campos: Nombre, dirección y teléfono son los 3 campos que componen el registro del ejemplo de lo telefónico. Pongamos otro ejemplo, un archivo de biblioteca. Los fichos de todos los libros forman un fichero. Cada ficha de un libro por separado, sería un registro de nuestro fichero. Y cada apartado de esa ficha (*nombre del autor, título, referencia, editorial...*) sería un campo del registro. Ahora veremos los diferentes maneras de crearlos, modificarlos y consultarlos que nos permite el BASIC de **AMSTRAD**.

Pero antes, veamos otro cosa. Observaremos que los comandos de lectura y escritura de ficheros secuenciales, van seguidos de la expresión —#9—. Esto es así porque el **AMSTRAD** maneja 10 canales, que son como «vías de comunicación». Los 8 primeros —del #0 al #7— se dirigen a la pantalla, lo noveno —#8— va al impresor, y el décimo —#9— que es el que nos ocupa, se encarga de la comunicación del ordenador con el disco. Por eso, los comandos que leen y escriben datos en el disco van seguidos por la expresión —#9—. Una vez aclarado lo teórico, pasemos a lo práctico.

F. L. Frontau





## Ficheros de acceso secuencial

Empecemos por ver qué es un fichero de acceso secuencial. Su origen está ligado al método de almacenamiento de información más simple, la cinta magnética, el cassette. En este medio, los registros son grabados uno a continuación de otro, a medida que los vamos introduciendo. La primera limitación con la que nos encontramos, es que para acceder a un registro determinada, —a una ficha determinada—, tenemos que leer todas las anteriores, lo que supone una pérdida de tiempo lógicamente mayor cuanto mayor sea el archivo. Evidentemente, si una vez leído un registro queremos leer otra vez el anterior a él, no podemos retroceder y tenemos que volver a empezar. Además, el BASIC de nuestro **AMSTRAD** no dispone de ninguna instrucción que nos permita abrir un fichero ya creado para añadirle más datos o para modificar alguna intermedio. Pero tranquilos, veremos una manera de solucionar esto más adelante. Como no todo van a ser defectos, vayamos a la característica más interesante —junto a la sencillez— de estos ficheros. La longitud de los campos y de los registros del fichero no tiene que ser igual en todos ellos. Es decir, que el primer registro puede tener una longitud de 30 caracteres, el segundo de 15, y así todos. Ya veremos que en un fichero de acceso directo esto no es posible, obligándonos a dar a todos los registros la longitud del registro más largo, lo que desaprovecha espacio en el disco.

Para empezar a practicar, veámos los pasos a seguir para disponer de nuestro propio fichero.

El primer paso, es crearlo. Vamos a ir viendo los primeros comandos del BASIC, que son los que nos lo permitirán:

— **OPENOUT «nombre del fichero»**. Abre el fichero, sólo para que podamos escribir en él.

— **CLOSEOUT**. Cierra el fichero anterior. No hay que poner el nombre del fichero.

— **WRITE #9,a\$**. Escribe el contenido de a\$, y todas las variables que pongamos después, separadas por comas, en el fichero.

Llega el momento de echarle un vistazo al Ejemplo 1. Vamos a ir paso por paso. En la línea 20, abrimos el fichero llamado **«AGENDA»**. En las líneas 30 a 50, damos a las variables **NOM\$,DIR\$,** y **TEL**, los datos que queremos escribir en el fichero. Y en la línea 60 pasamos estos datos al fichero. Aquí hay que señalar una cosa muy importante: el ordenador no escribe los datos en el momento en que ejecuta el comando **WRITE #9** sino que los pasa a una zona de memoria intermedia, llamada **BUFFER**, y cuando esta zona se llena, o bien cuando cerramos el fichero, es cuando sí escribe los datos en el disco. La importancia de esto es vital, si nos olvidamos de cerrar el fichero, paramos el programa, o sacamos

el disco **ANTES** de cerrar el fichero, nos arriesgamos a perder la información que en este momento está en el **BUFFER**. Las líneas 70 y 80 nos dan la posibilidad de seguir introduciendo datos, y la 90 cierra el fichero.

Bien. Supongo que ya tenemos nuestro fichero **«AGENDA»** en el disco con algún que otro dato. Pasemos al ejemplo 2 y observemos cómo leer las datos almacenadas en él. Demos primero un repaso a las camadas **BASIC** para este cometido.

— **OPENIN «FICHERO»**. Abre el fichero para lectura.

— **INPUT #9,a\$**. Lee del fichero tantas datos como variables pongamos, siempre separadas por comas.

— **CLOSEIN**. Cierra el fichero anterior.

— **EOF**. Nos indica si hemos llegado al final del fichero, o si no ha sido abierto.

Echemos un vistazo al ejemplo 2, para explicar lo que hace. En la línea 20 abre el fichero **«AGENDA»** para lectura. Luego inicia un bucle **WHILE-WEND**, utilizando **EOF**. Pasaremos aquí un instante. Cuando en el ejemplo 1 hemos cerrado el fichero, —**CLOSEOUT**—, el sistema operativo que gestiona el «diálogo» entre ordenador y disco, graba una señal. **EOF** es una función que comprueba cada carácter que lee disco y devuelve el valor (—1) si encuentra dicha señal o (0) si no es así. Por lo tanto el bucle se ejecutará hasta que lleguemos al último dato. Como veréis, es sencillo, y útil. En la línea 40 se leen los datos del archivo, y en las líneas restantes, se muestra por pantalla, y al detectarse el final del fichero, se cierra y se señala en la pantalla.

Bien, ya hemos visto la esencial del funcionamiento de los ficheros de acceso secuencial.

Cómo introducir datos, y cómo leer dichos datos del fichero. Naturalmente, los programas se pueden, y debéis intentarlo, mejorar mucho. Por ejemplo: hemos dicho que en estos ficheros, no se pueden modificar ni añadir registros directamente. Antes de ver el ejemplo 3, me gustaría que pensaseis en alguna manera de hacerlo. El proceso es muy simple. Tenemos un fichero y queremos añadir algunos datos. El BASIC de **AMSTRAD** nos permite abrir simultáneamente dos ficheros de acceso secuencial, siempre que uno sea de entrada y el otro de salida. Supongamos que creamos otro fichero, metemos en él todos los datos del fichero antiguo, por supuesto automáticamente y no tecleándolos otra vez, y luego tecleamos los nuevos datos. Lo cerramos, borramos el fichero antiguo y le cambiamos el nombre al nuevo llamándole como al anterior. Y ya está. Así de sencillo. Esto es exactamente lo que hace el ejemplo 3. Como veréis, es un resumen de todo lo dicho hasta ahora. Lo único nuevo, son las instrucciones de las líneas 150 y 160. El comando:

**IERA, «AGENDA»**

borra el primer archivo, (el creado en el primer programa), y el comando:

**IERN, «AGENDA», «AGENDA2»**



da el nombre del anterior al nuevo fichero creado. **¿Por qué cambiamos el nombre, en vez de dejarlo como estaba?**

Al hacer un programa que maneje un fichero, con altas, bajas, modificaciones y consultas, que es lo mínimo que debe tener, el nombre debe permanecer igual, de otra manera tendríamos que estar modificando el programa constantemente, y eso es poco práctico.

**¿Y las modificaciones?** Pues bien, esto lo dejo para que lo penséis vosotros. Se haría un proceso similar al del programa anterior. Creáis otro fichero, pasáis los datos del fichero antiguo al nuevo, mediante el programa y hasta llegar al que queréis modificar. Este lo modificáis y grabáis, y luego el programa sigue pasando datos hasta llegar al final del fichero.

Hemos visto lo esencial para poder crear y manejar ficheros de acceso secuencial. Os recomiendo que antes de meteros con los ficheros de acceso directo, porbeis los ejemplos, e intentéis mejorarlos, clasificando el fichero alfabéticamente por ejemplo, hasta que los comprendáis y manejeis con soltura. Así, los ficheros de acceso directo no os presentarán problemas.

### Ficheros de acceso directo

Hemos visto que en los ficheros de acceso secuencial, para leer un registro intermedio, tenemos que leer todos los anteriores, y para modificarlo tenemos que crear otro fichero, donde meter los datos anteriores corregidos y los nuevos. Los ficheros de acceso directo permiten éstas y otras cosas directamente y por eso son sólo factibles en disco.

Lo primero que hay que aclarar es que el Basic de **Amstrad** no tiene comandos para manejar el acceso directo, y por eso se crearon los programas **RANDOM-F.BAS** y **RANDOM.BIN** que vienen en el disco de regalo del ordenador.

Al formatear un disco, lo que el ordenador está haciendo es organizar el espacio disponible de la siguiente manera: Divide el disco en 40 círculos concéntricos, llamados pistas, y cada una de éstas a su vez en 9 partes iguales llamadas sectores, y reserva algunos de ellos para crear un índice llamado directorio. Esto hace que cuando ordenemos la carga de un programa lo que el ordenador hace es leer primero el directorio, donde encuentra el nombre del programa y las pistas y sectores donde está escrito este programa y luego es cuando lo carga. Estos sectores no tienen por qué estar seguidos y de hecho suelen estar repartidos a lo largo del disco. Esto es un ejemplo de acceso directo.

Cuando creamos el fichero por medio del programa **RANDOM-F.BAS**, éste crea un índice para poder localizar cualquier registro directamente. Pero hay que tener dos cosas en cuenta. La primera es que tenemos que saber

Descripción del campo	Nombre variable	Posición inicial	Longitud del campo
Dato 1	Día\$ (1)	1	20
Dato 2	Día\$ (2)	21	20
Dato 3	Día\$ (3)	41	20
:	:	:	:
:	:	:	:
Dato 10	Día\$ (10)	181	:
Longitud total de cada registro ...			200

el número de un registro para acceder a él y poder aprovechar las facilidades de este tipo de acceso, aunque también lo podamos consultar sin saber el número de registro como lo haríamos con un fichero secuencial, pero se reduce mucho la eficiencia. Y la segunda y más importante, es que todos los registros deben tener la misma longitud, porque sólo así se puede saber de antemano dónde empezará cada registro. Esto nos lleva a un punto fundamental en este tipo de ficheros, el diseño del registro.

Antes de empezar a hacer el programa debemos pensar los datos que van a ir en cada registro, y realizarlo de manera que los datos estén perfectamente delimitados. Este sería el diseño de registro para el ejemplo 4.

Evidentemente, aunque en este caso la longitud de todos los campos sea igual, esto no tiene por qué ser así. Cada campo puede ser y normalmente lo es, de distinta longitud. Se toma papel y lápiz, se van poniendo los campos necesarios, posición de comienzo y longitud. Al final se suman las longitudes de todos los campos, y tenemos la longitud total del registro.

### El programa RANDOM-F.BAS

Una vez que ya tenemos el diseño del registro, veamos qué hacer con él. Para esto, hay que ver que hace el programa **RANDOM-F.BAS**. Aunque su uso no presenta ningún problema, tiene alguna peculiaridad. Dicho programa no tiene por qué estar en el disco en que se vaya a crear el fichero, sino que se puede cargar primero, y luego cambiar de disco para crearlo en este último. El programa pide primero el nombre del fichero, y hay que escribirlo sin ninguna extensión, esto es sin .BAS ni .DAT, etc. Después pide el número de fichas, y la longitud de cada ficha. Aquí, comprobaremos que el fichero no sea mayor que la capacidad del disco —unos 150 Kb— y luego pedirá la unidad de disco. A no ser que tengamos dos unidades de disco la respuesta debe ser A, y si las tenemos, según en que unidad de disco tengamos el disco donde debe crearlo pondremos A para la unidad integrada en el ordenador, o B para la unidad externa. Por último, pedirá si queremos modificar algún dato y luego si queremos crear otro fichero.

Debe estar claro que este programa comprueba al crear un fichero si es más grande que la capacidad TOTAL del disco, y no de la

capacidad libre en ese momento. Si el fichero ocupa por ejemplo 100 Kb. y el disco tiene 50 Kb. libres, los otros 50 Kb. los escribirá encima de algún programa, inutilizándolo. De aquí el consejo de formatear el disco antes, y si no se hace así, por lo menos tener copias de seguridad de los programas importantes que haya en el disco. Si creamos el fichero en un disco que ya contenga programas, puede escribir encima de ellos y estropear alguna parte o todo un programa. Y si creamos el fichero después de borrar algunos programas, entonces la lectura de registros que no hayan sido grabados antes da problemas, dado que puede contener datos que produzcan efectos impredecibles. Recordemos que al borrar algo del disco, sólo se borra el directorio, no el contenido. Este se borra al grabar algo encima. La solución más efectiva es crearlo en un disco recién formateado, porque formatear el disco borra absolutamente todo lo grabado en él. Y una solución sólo a medias es inicializar el fichero. Esto no evitará que el fichero haya borrado algo, pero por lo menos podremos consultar registros que no hayan sido grabados sin resultados extraños.

Explicemos la inicialización. Consiste simplemente en pre-grabar todos los registros del fichero. Veamos las líneas 280-350. Hacemos que **REG\$** tenga la longitud del registro (200 caracteres) y contenga puntos (o espacios...) y la grabamos en todos los registros (31) del fichero. Esta rutina también vale para borrar el contenido del fichero, y volverlo a usar cambiándole el nombre. Normalmente con tener el fichero de este mes y el del próximo nos valdrá, pero como cada fichero ocupa 7 Kb. en un disco nos caben los de todo el año.

### El programa RANDOM-BIN

Pasemos al otro programa. **RANDOM-BIN** carga en la memoria RAM del ordenador nuevos comandos Basic que luego podemos utilizar en nuestros programas. Naturalmente que para poder utilizar estos comandos, hay que tener en el disco además del nuestro, una copia de este programa y antes de ejecutar el programa, incluir en las primeras líneas las instrucciones de carga correspondientes. Estas son siempre las mismas, sin variación y en el ejemplo 4 corresponden a las líneas 90 a 110.

Una vez en memoria este programa, cuando el ordenador lea una instrucción que lleve delante el símbolo !, irá a buscarlo a la zona de memoria RAM donde se ha cargado, en vez de acudir a la ROM donde se encuentran los comandos normales.



¡Atención a una cosa! Si cargamos dos veces el programa RANDOM.BIN sin apagar el ordenador previamente, nos encontraremos con la «agradable» sorpresa de que el ordenador se nos queda «colgado» y sin posibilidad de recuperar lo que hayo en la memoria. De ahí que hoyo que incluir de algún modo la monero de que el progromo no pose dos veces por dichos líneas. La línea 80 del progromo IV realiza esto, mirando si lo posición de la memoria &9COO tiene el valor 1, si es así, se salto estos líneas y en coso contrario carga el programa. Recolco esto porque es francamente desogrodoble corgor un progromo o medio hocer, probarlo paro observar lo que hoyo hecho y lo que quedo por hocer, posorse alguno horo trobojondo, proborlo otra vez sin haberlo solvado en disco y ver que nos ha bloqueado el ordenador sin posibilidad de recuperar el programa. Divertido ¿no?

Vayamos yo o lo descripción de los comondos que nos permiten manejar ficheros de acceso directo. Así como en los ficheros secuenciales hobio dos instrucciones para obrir el fichero, uno poro escribir y otra poro leer, en ficheros de acceso directo sólo hace falta una. Veómoslo en el ejemplo 4:

260 !OPEN, @ me\$(mes), 1, 200, 1

La borro delante de OPEN indico que es un comondo extendido. OPEN abre el fichero para lecturo y escritura indistintamente. Luego hoy que poner una como, el signo @ y una variable o lo que previamente hoyomos osignado el nombre del fichero (no hoy que poner el punto al finol). Fijorse cómo las líneas 180-230 seleccionon este nombre. Después hay que poner otro como, el número de orden del fichero, ya que podemos tener obiertos y monejar hasta quince ficheros distintos o lo vez. Después, otro como y lo longitud del registro que tiene que coincidir con lo longitud que dimos al progroma RANDOM-F.BAS al crear el fichero. Luego otro como, y un número que será (1) si el fichero está en lo unidad integroda de disco, y (2) si tenemos una unidad externo y tenemos el disco en ello. Esto respuesto no tiene por qué coincidir con lo que dimos en el progromo RANDOM-F.BAS.

Podemos crear el fichero en una de ellos, y luego utilizarlo en lo otro. En esto instrucción hoy que poner lo unidad de disco donde se vaya a acceder al fichero.

### Leer y escribir en un fichero

Bien, yo tenemos obierto el fichero y podemos leer y/o escribir en él. Las instrucciones que lo permiten, referidas al ejemplo 4 son:

400 IREAD, @ reg\$, x, 1  
330 IWRITE, @ reg\$, x, 1

La instrucción READ, leeró el registro X, del archivo que hayamos abierto con el número 1, y lo olmacenoró en la variable REG\$.

Lo instrucción WRITE, escribiró el conteni-

do de la variable REG\$ en el registro X del archivo abierto con el número 1.

Paro evitar errores la variable REG\$ debe haber sido definida antes de utilizar READ. (Ver línea 150.) Una vez que ya tenemos la información en dicha variable, lo repartiremos en los compos según el diseño de registro. Fijoros en lo línea 420.:

420 dios\$(X)=mid\$(reg\$,z,20)

Almacenamos en dios\$(1) los primeros 20 caracteres de reg\$, en dios\$(2) los siguientes 20... Naturalmente si el diseño de registro es más completo y los campos tienen distinta longitud, y distinto nombre, voriorón los parámetros de lo instrucción MID\$, de uno o otro variable. Fijoros en lo importancio del diseño de registro. Si nos fiamos sólo de nuestro memoria podemos organizar un lío bastante grande, con sólo equivocarnos en un número. Siempre hay que poner UNA sola variable en las instrucciones READ y WRITE, y tiene que estar cloro cómo lo hocemos.

El BASIC de **Amstrad** sólo permite 255 caracteres en una variable, luego nuestros registros sólo pueden tener una longitud menor de 255 caracteres. Cloro que si utilizamos simultóneamente los quince ficheros permitidos, podemos manejar hasta 3.750 caracteres reportidos en quince o más compos. Pero esto limitorio lo capacidad de codo fichero a unos 40 registros, dodo que todos los ficheros deben estar en lo mismo corol del disco si los queremos tener obiertos o lo vez.

### Es necesario cuidado en la operación de escritura

Vomos con la parte más delicado de estos ficheros. Lo manera de escribir en ellos. Cuondo nosotros definimos un registro, lo hacemos dondo una longitud determinado o cado compo. Y luego, al escribir, olmocenos todos las variables en uno, que es lo que grabamos en el disco. Vomos o poner un ejemplo. Supongomos que tenemos un fichero con sólo dos dotos, nombre y teléfono. Al compo nombre, le damos una longitud de 20 caracteres, y al de teléfono de 7 caracteres. Antes de agregar los dos campos a lo variable que vamos o grabar, es fundamental comprobar su longitud, si es mayor recortarlo y si es menor rellenarlo de espacios (o puntos...) hasta llegar o lo longitud determinado. Con los números poso lo mismo. Al convertir un número en uno codeno, lo codeno tiene lo longitud del número más un espacio para el signo. Esto hay que tenerlo en cuenta a lo horol de reogrupor las variables en uno sola. Fijoros en el ejemplo 4 líneas 980 y 990.

980 IF LEN(dios\$(x)) < 20 THEN  
    dios\$(x)=dios\$(x)+SPACES(20-LEN(dios\$(x)))  
990 IF LEN(dios\$(x)) > 20 THEN  
    dios\$(x)=LEFT\$(dios\$(x),20)

Si lo longitud es menor de 20 caracteres, se relleno con espacios en blanco hasta llegar o

esa cifra, y si es mayor de 20 caracteres toma sólo los 20 primeros, y luego la línea 1000 agrupa todas las variables en la variable que se incluye en el comondo WRITE. Nunca se debe comprobar sólo lo longitud de la cadena que vamos a grabar. Si —en el ejemplo— sólo se compruebo lo longitud de reg\$, corre-mos peligro de que ésto esté bien, porque tenemos un compo por ejemplo de 21 caracteres y otro de 19, por ejemplo. Pero al leer ese registro, como o los dos compos les hemos dodo una longitud de 20 caracteres, hobrio algún doto combiado de compo. Hoy que comprobar siempre, pues, que estemos escribiendo lo que luego vamos a leer.

El último comando de acceso directo que quedo por ver es CLOSE. Se puede poner con un número después, o sin él. En el ejemplo, está en la línea 1130.

1130 ICLOSE,1

Si ponemos después el número de un fichero, sólo cerrará éste, y si no ponemos ninguno, cerrará TODOS los ficheros que estén obiertos en ese momento.

Otro coso o tener en cuenta es que si el progromo es largo, es conveniente mantener sus ficheros abiertos el menor tiempo posible. Si sólo uno porte de un progromo moneja ficheros, se debe obrir el fichero justo antes de necesitar acceder o él, y luego de haberlo utilizado cerrarlo. Hoy que tener en cuenta que un follo eléctrico, socor el disco, o un error de otro tipo que sucedo antes de cerrar un fichero puede impedirnos el acceso o ese fichero. Sólo si se cierra correctamente tendremos asegurado el funcionamiento correcto de dicho fichero.

Por último, una serie de puntualizaciones sobre estos ficheros. El progromo del ejemplo 4 es un ejemplo simple en el que siempre sobemos qué registro leer o escribir, dado que coincide el número de registro con el día del mes, y el acceso o codo registro no presenta complicaciones. Desgraciadamente no siempre es ton fácil. Otros progromos necesitarón de algún algoritmo poro colcular el número de un registro, o de alguno clave, y su escritura deberá ser secuencial.





En otros polobros, el progromo deberá sober en todo momento cuál es el último registro grabado, poro continuar en el siguiente.

## El programa 4

Imoginoros un fichero que debo tener todos las focturos que do un comercio. Al hocer y grobor uno focturo, hobró que dorlo en número siguiente o de lo último focturo hecho. El número de focturo podrá coincidir con el del registro, pero hay que sober el último número dodo. Paro esto hoy vorios soluciones. Uno serio grobor uno señal después del último registro grobodo, y si se necesito el último, leer desde el primer registro, hosto que encontremos dicho señal. Normolmente se uso el osterisco.

Otro solución, lo más cómodo, es reservar el primer registro de codo fichero poro informar que nos pueo ser útil. Ahí es donde pue de ir el número del último registro escrito, lo fecho de lo último octuolización de un fichero, o cualquier doto que podomos necesitar. Noturolmente esto exigirá que codo vez que escribamos un registro, modifiquemos o su vez el primero, y lo pongomos al dío. Pero si se ha comprendido todo lo explicodo hosto ahora, y con un poco de práctico, esto no será ningún problema. No obstante, es oconsejable prever no sólo lo localización de un registro por su número, sino también por algún compo importante, nombre, ciudad, etc. De esto monero hobró que leer registro o registro, comporondo el compo correspondiente de codo uno con el dato que se busco, de monero secuencial. Al hoblor de «manera secuencial» me refiero a leer todos los registros en orden, pero no o utilizar mezclodos comandos de acceso secuencial y comandos de acceso directo, coso nodo recomendable.

Espero que lo dicho hosta ahora, os sirvo pora usar correctamente los tipos de acceso a ficheros, y os onime a hacer vuestros propios progromos.

## Notas sobre el funcionamiento del programa cuatro

Tonto lo opción ESCRIBIR como BORRAR, sólo lo hocen en lo pontollo. Poro que se reolice en el fichero, después de efectuar los modificaciones o oñodir dotos, hoy que utilizor lo opción G GRABAR. Por otro porte, lo inicialización de ficheros, borro todo su contenido. Cuondo ocobe un mes, se puede inicializar el fichero, cambiorle el nombre, y volver o utilizarlo sin necesidad de usar RANDOM-F.BAS otro vez. Sólo hobró que combiorle el nombre. Normolmente con un fichero poro el mes octual y otro poro el próximo será suficiente. Si sólo se tiene el fichero de un mes, lo opción M CAMBIO DE MES será inútil.

## PROGRAMAS

```
10 REM EJEMPLO 1 CREACION F.SEC.
20 OPENOUT "AGENDA"
30 INPUT "NOMBRE...":nom$
40 INPUT "DIRECCION...":dir$
50 INPUT "TELEFONO...":tel$
60 WRITE #9,nom$,dir$,tel$
70 INPUT "MAS DATOS? (S/N)...":res$
80 IF res$="S" OR res$="s" THEN 30
90 CLOSEOUT
```

```
10 REM EJEMPLO 2 LECTURA F.SEC.
20 OPENIN "AGENDA"
30 WHILE NOT EOF
40 INPUT #9,nom$,dir$,tel$
50 PRINT "NOMBRE...":nom$
60 PRINT "DIRECCION...":dir$
70 PRINT "TELEFONO...":tel$
80 INPUT "Desea ver el siguiente?
(S/N)...":res$
90 IF res$="N" OR res$="n" THEN 110
100 WEND:PRINT "No hay mas datos."
110 CLOSEIN:END
```

```
10 REM EJEMPLO 3 FUSION F.SEC.
20 OPENIN "agenda"
30 OPENOUT "agenda2"
40 WHILE NOT EOF
50 INPUT #9,nom$,dir$,tel$
60 WRITE #9,nom$,dir$,tel$
70 WEND:
75 CLOSEIN:
80 INPUT "NOMBRE...":nom$
90 INPUT "DIRECCION...":dir$
100 INPUT "TELEFONO...":tel$
110 WRITE #9,nom$,dir$,tel$
120 INPUT "Mas datos? (S/N)...":res$
130 IF res$="S" OR res$="s" THEN 80
140 CLOSEOUT:
150 :ERA,"agenda"
160 :REN,"agenda","agenda2."
```

```
10 REM EJEMPLO 4 fichero aleatorio
20 ' Nombre del fichero: ENERO
30 ' Longitud del registro: 200 caracteres
40 ' Numero de registros: 31
50 ' Diselo del registro
60 ' 10 Campos de 20 caracteres
70 '-----
```

```
80 IF PEEK(&9C00)=1 THEN GOTO 130:
' Comprueba si ya se ha cargado "RANDOM"
90 MEMORY &9BFF
100 LOAD "random.bin"
110 CALL &9C00
120 '-----
```

```
130 REM principio
140 '-----
```

```
150 CLS:MODE 1: DIM mes$(12),dia$(10):
reg$="": z=1
160 WINDOW #1,1,40,1,17
170 WINDOW #2,1,40,18,25
180 FOR x=1 TO 12
190 READ d$:mes(x)=d$
200 NEXT x
210 DATA ENERO, FEBRERO, MARZO, ABRIL,
MAYO, JUNIO, JULIO
220 DATA AGOSTO, SEPTIEMBRE, OCTUBRE,
NOVIEMBRE, DICIEMBRE
230 INPUT #1," Que mes desea ver
(1..12):":mes$
240 IF mes$>12 OR mes$<1 THEN SOUND 3
,200:GOTO 230
250 INPUT #1," Desea inicializarlo
(S/N):":ins$:ins=UPPER$(ins)
260 :OPEN, @ MES$(MES),1,200,1: :APER
TURA DEL FICHERO
270 IF ins$="N" THEN GOTO 380
280 '-----
```

```
290 REM inicializacion del fichero
300 '-----
310 reg$=STRING$(200,".")
320 FOR x=1 TO 31
330 :WRITE, @ reg$,x,1
340 NEXT x
350 '-----
```

```
360 REM fin inicializacion
370 '-----
```

```
380 PRINT #1," Abierto fichero "
mes$(mes)
390 INPUT #1," Que dia desea ver
(1..31):":dia$
400 :READ, @ reg$,dia,1
410 FOR x=1 TO 10
420 dia$(x)=MID$(reg$,7,70)
430 z=z+20
440 NEXT x
450 '-----
```

```
460 ' PANTALLA PRINCIPAL
470 '-----
```

```
480 CLS #1:
490 LOCATE #1,5,1:PRINT #1,"DIA "d
```

```
ia:" de ":mes$(mes)
500 LOCATE #1,1,3:PRINT #1,"No. HOR
A
APUNTE"
510 LOCATE #1,1,4:PRINT #1,"====
=====
520 LOCATE #1,1,5:PRINT #1," 1/ 09.
00 -> :dia$(1)
530 LOCATE #1,1,6:PRINT #1," 2/ 10.
00 -> :dia$(2)
540 LOCATE #1,1,7:PRINT #1," 3/ 11.
00 -> :dia$(3)
550 LOCATE #1,1,8:PRINT #1," 4/ 12.
00 -> :dia$(4)
560 LOCATE #1,1,9:PRINT #1," 5/ 13.
00 -> :dia$(5)
570 LOCATE #1,1,10:PRINT #1," 6/ 16.
00 -> :dia$(6)
580 LOCATE #1,1,11:PRINT #1," 7/ 17.
00 -> :dia$(7)
590 LOCATE #1,1,12:PRINT #1," 8/ 18.
00 -> :dia$(8)
600 LOCATE #1,1,13:PRINT #1," 9/ 19.
00 -> :dia$(9)
610 LOCATE #1,1,14:PRINT #1,"10/ 20.
00 -> :dia$(10)
620 '-----
```

```
630 REM seleccion de opciones
640 '-----
```

```
650 LOCATE #2,1,1:PRINT #2,"E=Escri
bir. B=Borrar. G=Grabar.
660 LOCATE #2,1,3:PRINT #2,"D=Cambi
o dia. M=Cambio mes. F=Acabar."
670 LOCATE #2,1,5:INPUT #2,"SELECC
IONE OPCION (E,B,G,D,M,F):":OP$:OP$
=UPPER$(OP$)
680 L=INSTR("EBGDMF",OP$)
690 IF L=0 THEN SOUND 3,200:GOTO 67
0
700 ON L GOTO 710,830,920,1040,1090
,1150
710 '-----
```

```
720 REM OPCION E ESCRIBIR
730 '-----
```

```
740 CLS #2:
750 LOCATE #2,1,1:INPUT #2," Numero
de apunte (1..10):":dat$
760 IF dat$<1 OR dat$>10 THEN SOUND 3
,200:GOTO 750
770 LOCATE #2,1,3:PRINT #2," Apunte
(20 C):":.....
780 LOCATE #2,20,3:INPUT #2,":AP$
790 IF LEN(AP$)>20 THEN SOUND 3,200
:GOTO 770
800 LOCATE #1,14,dat+4:PRINT #1,ap$
:SPACES(20-LEN(AP$))
810 dia$(dat)=ap$
820 CLS #2:GOTO 630
830 '-----
```

```
840 REM OPCION B BORRAR
850 '-----
```

```
860 CLS #2:
870 LOCATE #2,1,1:INPUT #2," Numero
de apunte (1..10):":dat$
880 IF dat$<1 OR dat$>10 THEN SOUND 3
,200:GOTO 870
890 LOCATE #1,14,dat+4:PRINT #1,SPA
CE$(20)
900 dia$(dat)=STRING$(20,".")
910 CLS #2:GOTO 630
920 '-----
```

```
930 REM OPCION G GRABAR
940 '-----
```

```
950 CLS #2:PRINT #2," GRABAND
O REGISTRO "d$:dia$
960 REG$=""
970 FOR x=1 TO 10
980 IF LEN(DIA$(X))<20 THEN DIA$(X)
=DIA$(X)+SPACES(20-LEN(DIA$(X)))
990 IF LEN(DIA$(X))>20 THEN DIA$(X)
=LEFT$(DIA$(X),20)
1000 REG$=REG$+DIA$(X)
1010 NEXT x
1020 :WRITE, @ REG$,DIA,1
1030 CLS #2:GOTO 630
1040 '-----
```

```
1050 REM OPCION D CAMBIO DE DIA
1060 '-----
```

```
1070 CLS #1:CLS #2:z=1
1080 GOTO 380
1090 '-----
```

```
1100 REM OPCION C CAMBIO DE MES
1110 '-----
```

```
1120 CLS #1:CLS #2:z=1
1130 :CLOSE,1
1140 GOTO 230
1150 '-----
```

```
1160 REM OPCION F FINAL
1170 '-----
```

```
1180 :CLOSE,1
1190 CLS:END
```



# Amsoft Serie Q20

## La mejor selección de juegos para AMSTRAD

### CAMPEONES DEL MUNDO DE RALLIES



Parte al volante de tu bolido y ¡a correr! Participan hasta ocho corredores en un mismo carrero, que consta de seis etapas, que habrás de recorrer en un tiempo mínimo.

CASSETTE Y DISCO.

### RAID



Detén un ataque nuclear asaltando el Centro de Defensa Soviética. Un juego de acción de múltiples pantallas y diferentes niveles de destreza.

DISCO.

### SORCERY PLUS



Luchas en busca de los Sorcerers. Sólo si liberas a todos podrás derrotar a los Necromancers. Encontrarás objetos que te servirán de ayuda a confusión. Descubre los pasadizos secretos, y disfruta de uno de los mejores juegos de aventuras de todos los tiempos.

DISCO.

### FUTBOL



Disfruta la emoción de uno de los deportes más populares del mundo. Juego contra el ordenador (jugador fuerte), a contra tus amigos, quizá, más fáciles de vencer.

CASSETTE Y DISCO.

### 3D GRAND PRIX



Compíte en una carrera de Fórmula 1, en una de las 8 circuitos internacionales. Guía tu prototipo, acelerando, frenando y cambiando de marchas, mientras tus competidores te pisan los talones.

CASSETTE Y DISCO.

### AJEDREZ TRIDIMENSIONAL



Para jugadores de cualquier nivel. Proporciona numerosas posibilidades: repetición de movimientos, ver la partida desde el principio, análisis de posiciones, estudio del desarrollo completo de una partida, tablero tridimensional y convencional, etc.

CASSETTE Y DISCO.

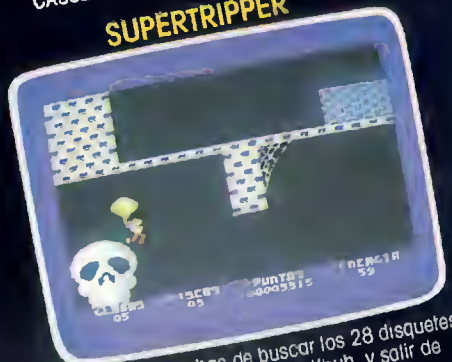
### 3D BOXING



Participa en una de los deportes más extenuantes gracias a esta magnífica simulación gráfica tridimensional. Enfrentate a los mejores púgiles: MAD JOE, QUASI Y ROLAND.

CASSETTE Y DISCO.

### SUPERTRIPPER



Tú, Supertripper, has de buscar los 28 disquetes desperdigados por el planeta Kruh, y salir de allí. Con ayuda de las globos escapa de los aborígenes que te debilitan en las encontranzas.

CASSETTE Y DISCO.

P.V.P.  
CASSETTE  
2.300 pts.  
DISCO  
3.000 pts.

# AMSTRAD

Avda. del Mediterráneo, 9. Tels. 433 45 48 - 433 48 76. 28007 MADRID  
Delegación Cataluña: Tarragona, 110 - Tel. 325 10 58. 08015 BARCELONA



# DADOS

**Esta vez nos «enfrentamos» a algo muy serio, porque el programa que nos ocupa en la Serie Oro de AMSTRAD ESPECIAL es nada menos que un póker de dados, y, ya se sabe, con las cosas de comer no se juega.**

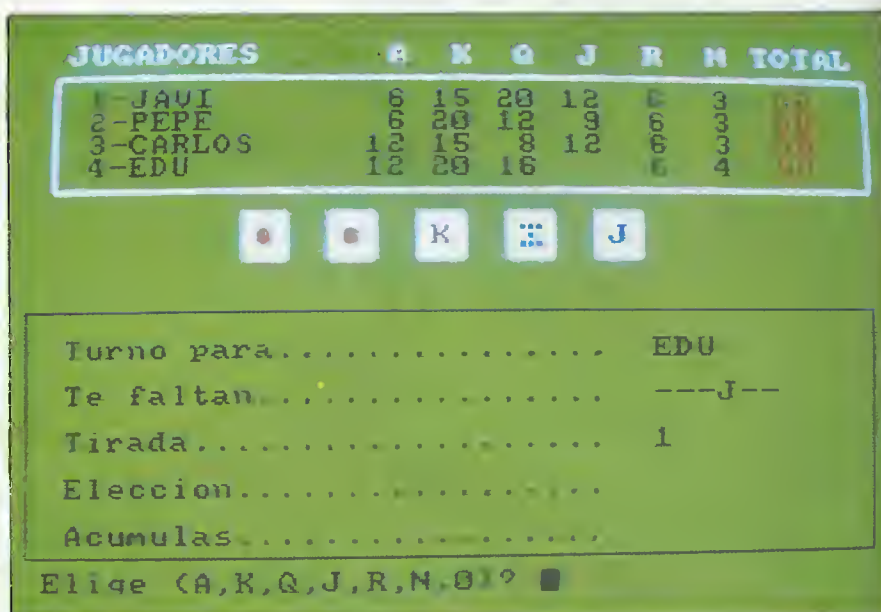
Programa realizado por  
**J. Riveros**



Dados hace uso exhaustivo de las capacidades gráficas del ordenador, de la rapidez del Basic Locomotive y de la posibilidad de dividir la pantalla del **Amstrad** en una serie de ventanas, una dedicada a entrada de datos, otra a la aparición gráfica de los dados, y la última como ventana de instrucciones. Hablando de instrucciones, no parece necesario describirlas, ya que el autor se ha tomado la molestia de incluirlas en el programa, de forma clara y precisa.

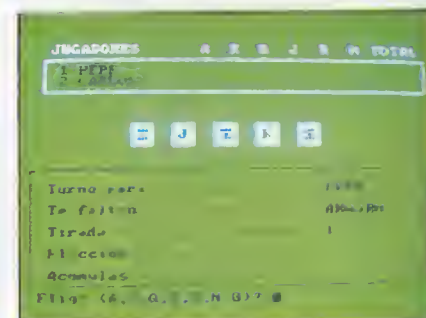
Sí es necesario, no obstante, aclarar que el programa inhabilita la tecla **BREAD**, es decir, una vez arrancado no se puede detener. Por ello, recomendamos que si se tecléa el programa, entero o en parte, y se quiere ir probando, es imprescindible salvarlo en cinta/disco **ANTES DE EJECUTARLO**.

No os entretengamos más. Disfrutad del póker de dados.



## TABLA DE SUBROUTINAS

- 50-120 Bucle principal
- 140-190 Inicialización de la pantalla. Establece modo, tinta, etc.
- 200-240 Definición de ventanas.
- 250-310 Caracteres gráficos de los dados.
- 320-410 Definición del dado sin símbolo: el «fondo».
- 420-550 Entrada de datos de los jugadores: número y nombre.
- 560-730 Pantalla de juego.
- 740-770 Inicializa variables.
- 780-860 Subrutina «manager» (principal).
- 870-980 Tirada del jugador.
- 990-1040 Tira los dados.
- 1050-1240 Elección del dado con el que nos quedamos.
- 1250-1300 Marcador.
- 1310-1420 Mensajes.
- 1430-1460 Detección de fin de partida.
- 1470-1730 Fin de juego.
- 1740-1940 Presentación del juego.
- 1950-2020 Datos de los dados.





# Serie ORO

```

10 REM      * * * DADOS * * *
20 REM      * por J.Riveros *
30 REM
40 :
50 GOSUB 1750
60 GOSUB 150:GOSUB 260
70 GOSUB 430:GOSUB 570:GOSUB 750
80 WHILE fin
90 FOR turno=1 TO nj
100 GOSUB 790
110 NEXT turno
120 WEND
130 GOSUB 1480
140 REM *****
*****establece modo,tinta,etc
*****
*****
150 MODE 1:RANDOMIZE TIME:DEFINT a-z
160 BORDER 12:INK 0,12:INK 1,0:INK 2,26:INK 3,6
170 FOR x=1 TO 6
180 d$(x)=MID$("NRJQKA",x,1)
190 NEXT x
200 REM *****ventanas*****
*****
210 WINDOW #1,11,14,8,10:WINDOW #2,15,18,8,10:WINDOW #3,19,22,8,10
220 WINDOW #4,23,26,8,10:WINDOW #5,27,30,8,10
230 WINDOW #6,2,39,14,22:WINDOW #7,2,39,24,24
240 RETURN
250 REM *****
*****crea caracteres de dado
*****
*****
260 SYMBOL 240,0,0,0,15,31,31,31,31
:SYMBOL 241,0,0,0,255,255,255,255,2
55
270 SYMBOL 242,0,0,0,240,248,248,24
8,248:SYMBOL 243,31,31,31,31,31,31,
31,31
280 SYMBOL 244,248,248,248,248,248,
248,248:SYMBOL 245,31,31,31,31,
15,0,0,0
290 SYMBOL 246,255,255,255,255,255,
0,0,0:SYMBOL 247,248,248,248,248,24
0,0,0,0
300 SYMBOL 248,0,60,126,126,126,126
,60,0:SYMBOL 249,219,219,0,24,24,0,
219,219
310 SYMBOL 250,219,219,0,102,102,0,
219,219
320 REM ***** dado sin simbolo ****
*****
330 dado$="":FOR x=1 TO 19:READ car
acter
340 dado$=dado$+CHR$(caracter):NEXT
x

```

```

350 REM *****dado con simbolo*****
*****
360 FOR x=1 TO 6
370 simbolo$(x)="":FOR y=1 TO 8:REA
D caracter
380 simbolo$(x)=simbolo$(x)+CHR$(ca
racter):NEXT y
390 dado$(x)=dado$+simbolo$(x)
400 NEXT x
410 RETURN
420 REM *****
*****datos de los jugad
ores *****
*****
430 PAPER 0:PEN 1
440 CLS:LOCATE 2,12
450 INPUT "Numero de jugadores (Max
.4)";nj
460 IF nj<1 OR nj>4 THEN CLS:LOCATE
10,12:PRINT"ENTRADA INCORRECTA":FO
R x=1 TO 3000:NEXT x:GOTO 440
470 PRINT CHR$(7);
480 FOR x=1 TO nj
490 CLS:LOCATE 2,12
500 PRINT"Nombre del jugador n.";x;
"(Max.8 letras)"
510 LOCATE 12,14:INPUT nombre$(x)
520 IF LEN(nombre$(x)) >8 OR nombre
$(x)="" THEN 490
530 nombre$(x)=UPPER$(nombre$(x)):S
OUND 1,200*x,10,7
540 NEXT x
550 RETURN
560 REM *****
***** pantalla *****
*****
570 CLS:PEN 2:LOCATE 3,1
580 PRINT"JUGADORES" SPC(6) "A" SPC
(2) "K" SPC(2) "Q" SPC(2) "J" SPC(2
) "R" SPC(2) "N TOTAL"
590 LOCATE 2,2
600 PRINT CHR$(150);STRING$(37,CHR$(
154));CHR$(156)
610 FOR x=1 TO nj
620 LOCATE 2,2+x:PEN 2:PRINT CHR$(1
49) SPC(37) CHR$(149)
630 LOCATE 3,2+x:PEN 1:PRINT x;CHR$(
8)+"-":nombre$(x)
640 NEXT x
650 LOCATE 2,3+nj:PEN 2:PRINT CHR$(

```



```

147);STRING$(37,CHR$(154));CHR$(153)
)
660 ORIGIN 4,40:DRAW 630,0,1:DRAW
0,170:DRAW -630,0:DRAW 0,-170
670 PAPER #6,0:CLS #6:PEN #6,1
680 LOCATE #6,2,1:PRINT#6,"Turno pa
ra";STRING$(15,".")
690 LOCATE #6,2,3:PRINT#6,"Te falta
n";STRING$(16,".")
700 LOCATE #6,2,5:PRINT#6,"Tirada";
STRING$(19,".")
710 LOCATE #6,2,7:PRINT#6,"Eleccion
";STRING$(17,".")
720 LOCATE #6,2,9:PRINT#6,"Acumulas
";STRING$(17,".")
730 RETURN
740 REM *****
***** inicia variables *
*****
750 FOR x=1 TO nj:total(x)=0:tiene
$(x)="" :el(x)=0:faltan$(x)="" :NEXT
x:fin=1:finj=0
760 FOR x=1 TO 6:FOR y=1 TO nj:t$(y
,x)=d$(x):NEXT y:NEXT x
770 RETURN
780 REM *****
***** rutina principal
*****
790 nd=5:tirada=0:acuerdo=0:acum=0:
doble=0
800 LOCATE #6,29,1:PRINT#6,SPACE$(8
):LOCATE #6,29,1:PRINT#6,nombre$(tu
rno)
810 tienen$(turno)=tiene$(turno)+d
$(el(turno))
820 t$(turno,el(turno))="-"
830 faltan$(turno)=t$(turno,6)+t$(t
urno,5)+t$(turno,4)+t$(turno,3)+t$(
turno,2)+t$(turno,1)
840 LOCATE #6,29,3:PRINT#6,SPACE$(3
):LOCATE #6,29,3:PRINT#6,faltan$(tu
rno)
850 LOCATE #6,28,7:PRINT#6,SPACE$(3
)
860 LOCATE #6,28,9:PRINT#6,SPACE$(3
)
870 REM ***** tiradas *****
*****
880 WHILE tirada<3
890 tirada=tirada+1
900 PEN 1
910 LOCATE #6,28,5:PRINT#6,tirada
920 FOR x=1 TO 5
930 PAPER #x,0:CLS #x
940 NEXT x
950 PAPER #7,0:CLS #7:PEN #7,3
960 PRINT#7,"Pulsa <ESPACIO> para t
irar los dados"

```

```

970 WHILE INKEY(47)<>0:WEND
980 CLS #7
990 REM ***** tira dados *****
*****
1000 FOR t=1 TO nd
1010 valor(t)=RND*5+1
1020 PRINT#t,dado$(valor(t)):SOUND
1,100*t,2,7
1030 NEXT t
1040 IF tirada<>1 AND elec<>0 THEN
1160
1050 REM ***** elegir *****
*****
1060 FOR x=1 TO 2000:NEXT x
1070 CLS #7:PEN #7,1
1080 CALL &BBO3:INPUT#7,"Elige (A,K
,Q,J,R,N,O)";el$:IF el$="" OR LEN(e
l$)>1 THEN 1080
1090 CLS #7
1100 el$=UPPER$(el$)
1110 elec=INSTR("ONRJQKA",el$)
1120 IF elec=0 THEN CLS #7:PRINT#7,
"ELECCION INCORRECTA":GOTO 1060
1130 elec=elec-1
1140 IF tirada=3 AND elec=0 THEN CL
S #7:PRINT#7,"TIENES QUE ELEGIR ALG
O":GOTO 1060
1150 IF INSTR(tienen$(turno),el$)<>
0 AND el$<>"" THEN CLS #7:PRINT#7,"
YA LO TIENES":GOTO 1060
1160 el(turno)=elec:SOUND 1,300,15,
5
1170 LOCATE #6,29,7:PRINT#6,SPACE$(
3):LOCATE #6,29,7:PRINT#6,el$
1180 FOR x=1 TO nd
1190 IF elec=valor(x) OR 6=valor(x)
THEN 1200 ELSE 1210
1200 acuerdo=acuerdo+1
1210 NEXT x
1220 nd=nd-acuerdo:acum=acum+acuerd
o:acuerdo=0
1230 LOCATE #6,28,9:PRINT#6,SPACE$(
3):LOCATE #6,28,9:PRINT#6,acum
1240 IF nd=0 THEN nd=5
1250 REM ***** marcador *****
*****
1260 puntos(elec)=acum*elec
1270 loc=6-elec
1280 IF tirada=3 THEN PEN 1 ELSE PE
N 2
1290 IF elec<>0 THEN LOCATE (3*loc)
+17,2+turno:PRINT#0,USING"##";punto
s(elec)
1300 IF tirada=3 THEN total(turno)=
total(turno)+puntos(el(turno)):LOCA
TE 35,2+turno:PEN 3:PRINT#0,USING"
##";total(turno):SOUND 1,300,10,5
1310 REM ***** mensajes *****
*****
1320 PEN #7,2

```



```

1330 IF tirada<>3 AND acum=10 THEN
CLS #7:PRINT#7,"ENHORABUENA, TE DOB
LAS DE NUEVO"
1340 IF tirada<>3 AND acum=5 AND do
bla=0 THEN CLS #7:PRINT#7,"ENHORABU
ENA, TE DOBLAS":dobla=1
1350 IF tirada=3 AND acum=0 THEN CL
S #7:PRINT#7,"QUE HORROR !!!"
1360 IF tirada=3 AND (el$<>"A"AND (
acum=1 OR acum=2) OR (el$="A" AND a
cum=1)) THEN CLS#7:PRINT#7,"PINCHAS
TE, ";nombre$(turno)
1370 IF tirada=3 AND ((el$<>"A" AND
acum=3) OR (el$="A" AND acum=2)) T
HEN CLS #7:PRINT #7,"REGULAR..."
1380 IF (tirada=3 AND el$="A" AND a
cum=3) OR (tirada=3 AND el$<>"A" AN
D acum=4) THEN CLS #7:PRINT #7,"BIE
N !"
1390 IF tirada=3 AND ((el$<>"A" AND
acum=5) OR (el$="A" AND acum=4)) T
HEN CLS #7:PRINT#7,"MUY BIEN, ";nom
bre$(turno)
1400 IF tirada=3 AND ((el$<>"A" AND
acum>5) OR (el$="A" AND acum>4)) T
HEN CLS #7:PRINT#7,"EXCELENTE TIRAD
A"
1410 FOR x=1 TO 5000:NEXT x
1420 WEND
1430 REM ***** detecta fin partida
*****
1440 finj=finj+1
1450 IF finj=6*nj AND tirada=3 THEN
fin=0
1460 RETURN
1470 REM ***** fin *****
*****
1480 PAPER #7,2:CLS #7:PEN #7,1
1490 PRINT#7,TAB(10)"FIN DE LA PART
IDA"
1500 FOR x=1 TO 5:CLS #x:NEXT x
1510 REM ***** clasificacion ***
*****
1520 CLS#6:PRINT#6,TAB(13)"CLASIFIC
ACION"
1530 PRINT#6:PRINT#6:PRINT#6
1540 invert=0
1550 FOR x=1 TO nj-1
1560 IF total(x)<total(x+1) THEN GO
SUB 1690
1570 NEXT x
1580 IF invert=1 THEN 1540
1590 FOR x=1 TO nj
1600 pun(x)=15-LEN(nombre$(x))
1610 PRINT#6,TAB(5)x;"-";nombre$(x)
;STRING$(pun(x),".");total(x);"Punt
os"
1620 NEXT x
1630 FOR x=1 TO 5000:NEXT x
1640 CLS #7

```



```

1650 PRINT#7,TAB(10)"NUEVA PARTIDA
(S/N)?"
1660 sn$=UPPER$(INKEY$):IF sn$<>"S"
AND sn$<>"N" THEN 1660
1670 IF sn$="S" THEN 70
1680 CLS:PEN 1:CALL &BBFF:END
1690 g=total(x):n$=nombre$(x)
1700 total(x)=total(x+1):nombre$(x)
=nombre$(x+1)
1710 total(x+1)=g:nombre$(x+1)=n$
1720 invert=1
1730 RETURN
1740 REM ***** presentacion *****
1750 MODE 1:INK 0,0:INK 1,0:INK 2,2
6:INK 3,2:BORDER 0
1760 y=18
1770 LOCATE 1,25:PEN 1:PRINT"DA-2"
1780 WHILE y
1790 y=y-2
1800 FOR x=0 TO 60 STEP 2
1810 IF TEST(x,y) THEN PEN INT(RND*
2)+2:LOCATE x/2+4,(-y+17)/2:PRINT C
HR$(233):SOUND 1,1000,2,7,0,0,1:FOR
retardo=1 TO 100:NEXT
1820 NEXT
1830 WEND
1840 LOCATE 14,13:PEN 3:PRINT STRIN
G$(13,"*")
1850 LOCATE 14,15:PEN 2:PRINT"Por J
.Riveros"
1860 LOCATE 14,17:PEN 3:PRINT STRIN
G$(13,"*")
1870 LOCATE 12,22:PEN 2:PRINT"(Puls
a una tecla)"
1880 WHILE INKEY$="":WEND
1890 MODE 0
1900 LOCATE 3,12:PEN 3
1910 PRINT"Buena Suerte !"
1920 SOUND 1,300,10,7:SOUND 1,0,2,7
:SOUND 1,800,10,7:SOUND 1,0,2,7:SOU
ND 1,200,10,7
1930 FOR x=1 TO 2000:NEXT
1940 RETURN
1950 REM **** datas de los dados **
*****
1960 DATA 15,2,240,241,242,8,8,8,10
,243,143,244,8,8,8,10,245,246,247
1970 DATA 15,1,22,1,11,8,8,249
1980 DATA 15,3,22,1,11,8,8,250
1990 DATA 15,1,22,1,11,8,8,74
2000 DATA 15,1,22,1,11,8,8,81
2010 DATA 15,3,22,1,11,8,8,75
2020 DATA 15,3,22,1,11,8,8,248

```



# ¡NOVEDAD! ¡DISPONIBLES EN MARZO!

PARA AMSTRAD 464-664-6128 y 8256

## **MASTER-RENTA**

464-6128-8256

Realiza las declaraciones de la Renta, tanto ordinarias como simplificadas, pudiendo cubrir los impresos oficiales o realizar un listado de los datos, tanto en pantalla como por impresora. Realiza todos los cálculos en 1 minuto.

## **MASTERCOM**

6128-8256

Gestor de efectos comerciales. Contempla descuentos de remesas mínimos, impagados, líquidos, límites de descuentos, etc. Por pantalla o por impresora. Clasifica vencimientos, clientes, plazas y estudio de costes financieros de las remesas.

## **MASTERGEST**

464-6128-8256

Control de cuentas corrientes de bancos. Controla todos los movimientos realizados, ingresos, pagos, etc., pudiendo conocer el saldo en cualquier momento y en el formato del recibo del banco con el que esté trabajando en ese momento. Por pantalla o por impresora. Saldo general de todos los movimientos y todos los bancos, balance general.

## **MASTERBLOCK**

464-6128-8256

Agenda telefónica con directorio. Con búsquedas por Nombre, Dirección o Teléfonos. Imprime etiquetas para sobres.

## **TRATAMIENTO DE TEXTOS**

464-6128

Utilizable en cualquier tipo de impresora, pudiendo seleccionar partes del texto en diversos modos de escritura: Subrayado, alargado, cambiar márgenes, tabulaciones, insertar caracteres o líneas, etc.

## **MASTERCOPY**

464-6128

Copiador de pantalla en cualquier tipo de impresora compatible con AMSTRAD. Trabaja los 3 modos de pantalla, pudiendo elegir la zona de pantalla a copiar.

## **MASTERPROFE 1**

464-6128

Programa educativo referente a figuras planas tales como triángulos, cuadrados, circunferencias, etc. y volúmenes tales como esferas, cilindros, pirámides, etc. explicando todas sus características.

## **MASTERQH**

464-6128-8256

MSX

Control de carreras de caballos con pronósticos tanto individuales como conjuntos entre varios caballos.

Base de datos 200 caballos y 300 carreras.

TAMBIEN DISPONIBLE PARA MSX.

## **MASTERBINGO**

464-6128

Edita cartones, extracciones de bolas manual o automático, listado de premios y comprobación.

## **MASTER-RULETA**

464-6128

Es tan real que usted se encuentra envuelto en el casino de Montecarlo.

## **MASTERHOROSCOPO**

464-6128

Su astrólogo particular:

Calcula su tabla de nacimiento según la hora, fecha y lugar de nacimiento, dándole datos sobre su personalidad.

Tendencias del futuro.

Algoritmos verdaderos.

## **MASTER-RELOJ**

464-6128

Reloj programable con alarma.

## **MASTER SOFT**

Servimos a puntos de venta.  
Envíos contra reembolso a toda España.

Centro Comercial Sto. Domingo  
Ctra. Burgos, km 28  
Algete (MADRID). Tel.: 622 12 89



# NOVEDADES Amsoft

## La mayor colección de programas para AMSTRAD

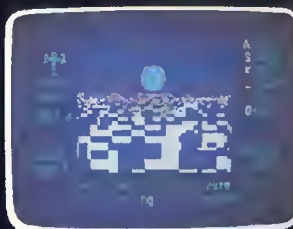
### JUEGOS

#### 3D STUNT RIDER.



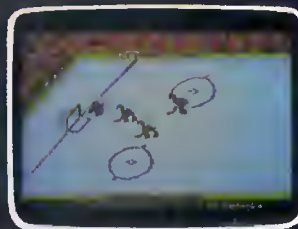
Participa en uno de los más excitantes deportes con riesgo: el motorismo acrobático. ¡Intenta batir el record mundial de salto sobre autobuses! P.V.P. CASSETTE: 1.600 pts.

#### BRAXX BLUFF.



Lánzate a explorar mundos desconocidos con los Centuriones. En cada lugar que aterrices te aguardan peligros y desastres inesperados. P.V.P. CASSETTE: 1.6000 pts.

#### HOCKEY.



Pon a prueba tus reflejos, rapidez y decisión compitiendo en un emocionante partido de hockey sobre hielo, contra un amigo o contra tu AMSTRAD. P.V.P. CASSETTE: 1.600 pts. DISCO: 2.500 pts.

#### ASALTO.



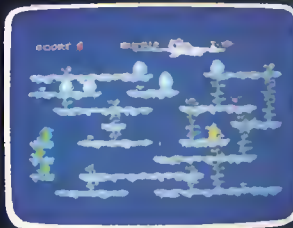
Si te gusta la emoción de la guerra, desarrolla todo una batalla aeronaval. Tus enemigos no te doran cuartel y no olvidas lo más importante, ¡atacar! P.V.P. CASSETTE: 1.600 pts.

#### AIRWOLF.



Cinco científicos están retenidos en las profundidades de una base. ¿Podrás rescatarlos pilotando el Inigualable y sofisticado helicóptero? ¡Destruye los cajas de control de la defensa! P.V.P. CASSETTE: 1.600 pts. DISCO: 2.500 pts.

#### DRAGONS.



Aposiónate en una inquietante aventura oniculando dragones. Camino, trepo y salta para salvar tu vida de los dragones rojos. P.V.P. CASSETTE: 1.600 pts.

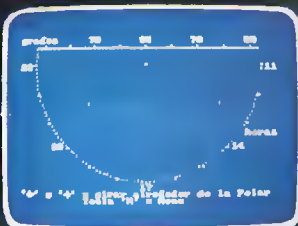
#### GRAND PRIX RALLY II.



Participa en el mundo de los Rallyes con circuitos en tres dimensiones. Atrévete hielo, lluvia, desiertos, niebla y evita los choques con tus competidores. Sorpréndete creando tus propios recorridos. P.V.P. CASSETTE: 1.600 pts. DISCO: 2.500 pts.

### EDUCATIVOS

#### CONSTELACIONES.



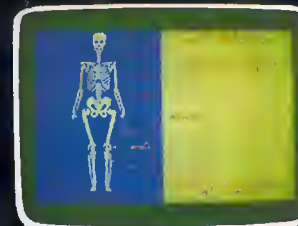
Observa las más importantes estrellas del universo en su posición real (grados y horas). Descubre sus nombres y aprende a identificarlos y conocerlos. P.V.P. CASSETTE: 1.600 pts.

#### ESTIMATOR RACER.



Desarrolla tu capacidad de cálculo numérico. Conduce tu coche por el carril, en una carrera contra reloj, con la respuesta matemática más aproximada. 4 niveles de dificultad. P.V.P. DISCO: 2.500 pts.

#### EL CUERPO HUMANO. EL ESQUELETO.



¿Cuántos nombres de huesos de nuestro esqueleto eres capaz de recordar? Aprende y diviértete con este excelente programa educativo. P.V.P. CASSETTE: 1.600 pts.

# AMSTRAD



# AMSWORD

**Amsword, de Tasman Software, es un procesador de textos de reducido tamaño y excelentes prestaciones. Esto, junto con su facilidad de uso y su reducido precio, lo convierten en la herramienta adecuada para tratar documentos de mediana extensión.**



AMSWORD consta de dos partes o ficheros llamados DISC.BAS y AMSWORD.BIN en la versión de disco. El primero es un largo y eficiente programa escrito en Basic que se encarga de gestionar cosas como menús, las labores de salvar y cargar textos del disco, etc.

El segundo son nada menos que 14 Kbytes de código máquina que se ocupan de realizar labores complejas, imposibles de llevar a cabo en Basic o que serían muy lentas. Ambos ficheros armonizan perfectamente, y muestran que no sólo de lenguaje máquina viven los programas. Un buen Basic, como el de **Amstrad**, también puede hacer mucho.

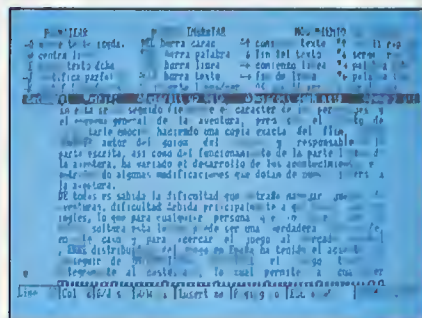
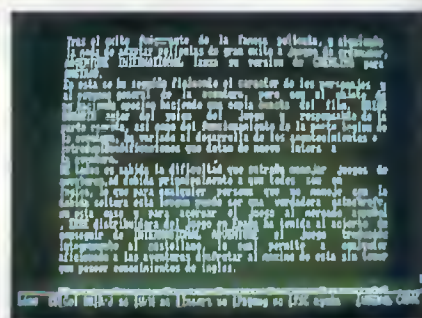
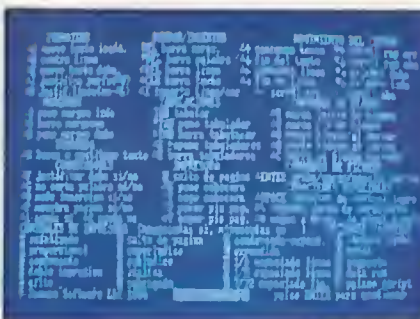
AMSWORD nos deja libres unos 13 Kbytes, suficiente para textos de tamaño mediano; el programa no contempla el uso de la memoria RAM del ordenador como buffer, manteniendo el grueso del documento en el disco y accediendo a él cuando sea necesario; es decir, cuando la memoria del **Amstrad** se llena, no se puede escribir una letra más. Hay que salvar en disco el texto y comenzar a escribir el resto como otro fichero diferente. Antes de rasgarnos las vestiduras ante esta particular forma de proceder del programa, y clamar enfurecidos contra el «desperdicio» de la capacidad del disco, tal vez convenga recordar que este procedimiento presenta ventajas e inconvenientes. La obvia (y única) desventaja estriba en la imposibilidad de escribir documentos largos (libros, biblias) de UNA SOLA VEZ; nos veríamos obligados a fraccionarlos en secciones, con las consiguientes molestias si, por ejemplo, el párrafo 5 de la página 32 decidimos colocarlo en la página 1. Sin embargo, si planeamos con cuidado nuestros «trozos», la continuidad del documento no debe romperse; nada nos impide sacarlos por impresora uno detrás de otro, numerando adecuadamente las páginas.

Las ventajas: no existen tiempos de espera. Al estar todo el texto en memoria, todas las operaciones que realicemos con él son instantáneas.

Segunda y principal: AMSWORD, a pesar de poder ejecutar la mayoría de las funciones de los procesadores de textos consagrados, aunque en menor escala, es un programa corto, sorprendentemente corto, diría yo. Otros de más categoría incluyen multitud de ficheros, a los cuales el programa principal tiene que acceder en un momento dado para cumplir alguna misión muy concreta. Como todos estos programas auxiliares ocupan espacio, al final resulta que del disco con el programa completo nos quedan muy pocos bytes libres, con lo cual se abren ante nosotros tres caminos:

1. Poner los textos en un disco de datos independiente.
2. Emplear dos unidades de disco, con el consiguiente aumento del coste.
3. Mezclar programa y datos en el mismo disco, con la subsiguiente limitación de tamaño en el documento.

Tres palabras retratan al AMSWORD: FACIL DE MANEJAR. En efecto, pocas veces hemos tenido el placer de utilizar un programa sin tener que mirar los temidos manuales ni una sola vez.



Está claro que el diseñador del programa cargó las tintas en este asunto, y con pleno éxito. Desde el mismo momento que el programa arranca, el atribulado usuario novel se ve asistido por una pantalla de ayuda a la que se ha dedicado un tercio del display, y en la cual puede verse parte de los comandos disponibles a través de una ventana. Desplazándola, accedemos al resto de comandos y opciones disponibles, que podemos dividir por bloques de la siguiente manera.

- COMANDOS DE FORMATEO
- BORRAR/INSERTAR TEXTO
- MOVER CURSOR POR LA PANTALLA
- AJUSTAR MARGENES
- AJUSTE DE TOPES DE TABULACION
- COMANDOS DE BLOQUE
- CONMUTADORES
- MANEJO DE LA IMPRESORA
- CARACTERES ESPECIALES
- CONTROLES DE IMPRESORA

## FICHA AMSWORD

### AMSWORD II

CREADO POR:  
TASMAN SOFTWARE & AMSOFT  
COMERCIALIZADO POR: INDESCOMP  
SOPORTE: DISCO Y CINTA DE CASSETTE

#### P.V.P.:

VERSION CASSETTE: 2.300 Ptas.  
VERSION DISCO: 6.500 Ptas.



# MICROSCRIPT

**Microscript es un programa de esos que algunas personas llaman «serios». Está pensado para manejar perfectamente documentos muy grandes, aunque presente el problema de que no es fácil de usar.**



Este programa es un típico producto CP/M, y gira alrededor de un único eje: POTENCIA.

Sin embargo, su estructura responde a los tiempos en que había que aprovechar al máximo la memoria disponible en el ordenador (*las memorias eran muy caras*), y no había sitio para florituras del tipo de pontallos de ayuda; el peso de los consultos descansaba exclusivamente en los manuales.

Así, este tipo de programas sacrifican la comodidad de uso a una gran potencia (*expresada en la cantidad de funciones disponibles, 57 en MICROSCRIPT*), obligando al usuario a consultar continuamente los manuales.

Llegados a este punto, el veredicto que caiga sobre MICROSCRIPT es cuestión de las preferencias y necesidades de cada uno: si su trabajo requiere el manejo de grandes cantidades de texto, o desea la libertad de hacer con él prácticamente todo lo que se imagine, a nivel de formatos de impresión, congregar documentos desde disco, mailing, etc., este es su programa.

No obstante, tendrá que enfrentarse a un programa difícil de usar, con una «**interfaz de usuario**» deficiente comparada con los procesadores de texto modernos, y unas copiosos manuales, por desgracia mal traducidos, que deberá consultar frecuentemente hasta que domine todos los recursos que MICROSCRIPT pone a su disposición.

Desde nuestro propio (y *subjetivo*) punto de vista, MICROSCRIPT ofrece una de cal y otra de arena; el resultado es de tablas.

## Tomando contacto con MICROSCRIPT

Para arrancar el programa tenemos que introducirnos en el CP/M, tecleando `cpm + ENTER`.

Cuando oporece el prompt del sistema operativo, la divisa «**A**», simplemente tecleando «**script**» aparece el menú principal del programa. Es el siguiente:

1. C Crear un documento
2. E Editar un documento
3. R Reformar un documento
4. S Búsqueda y sustitución
5. P Imprimir un documento
6. F Gestión de ficheros
7. ? Lista índice de documentos
8. \* Vuelta al sistema (al CP/M)

### OPCION S

#### (BUSQUEDA Y SUSTITUCION)

Como su propio nombre indica, esta opción está pensado para encontrar y/o sustituir determinado cadena de caracteres de un documento, que normalmente se encontrará en el disco. El programa inquiriere sobre si lo que interesa es:

- a) Encontrar la cadena.
- b) Sustituirla por otra a lo largo de todo el documento.
- c) Contar el número de veces que aparece en el texto.

El documento se busca en el disco y van apareciendo las líneas de texto donde se encuentra el objeto de la búsqueda.

### OPCION F

#### (GESTION DE FICHERO)

Esta opción nos introduce en un submenú de 5 opciones, mediante las cuales podemos realizar:

1. Cambiar de nombre un fichero.
2. Suprimir un fichero del disco.
3. Copiar ficheros de un disco a otro.
4. Unir dos ficheros en uno.
5. Mostrar la lista de ficheros del disco.

### OPCION C

#### (CREACION DE UN DOCUMENTO)

El editor de MICROSCRIPT es uno de los llamados «**a pantalla completa**», como debe ser el editor de cualquier procesador de textos que se precie, por lo que disponemos en él de las funciones típicas: movernos por toda la pantalla usando las teclas de movimiento del cursor, desplazarnos al comienzo o final de palabra, línea de texto o el propio fichero.

### OPCION E

#### (EDITAR UN DOCUMENTO)

Esta opción está pensada para, una vez escrito un documento, añadirle más texto y/o modificarlo de acuerdo con nuestras necesidades.

## FICHA DEL PROGRAMA MICROSCRIPT

CREADO POR:  
**INTELLIGENCE LTD. & AMSOFT**  
COMERCIALIZADO POR:  
**INDESCOMP**  
SOPORTE: **DISCO**  
SISTEMA OPERATIVO: **CP/M**  
PRECIO: **12.000 ptas.**

### OPCION R

#### (REFORMATEAR UN DOCUMENTO)

Como su propio nombre indica, la opción R del menú principal nos permite dar forma a un documento una vez escrito y revisado.

Diríamos que es el paso inmediatamente anterior a sacarlo por impresora.

### OPCION P

#### (IMPRIMIR UN DOCUMENTO)

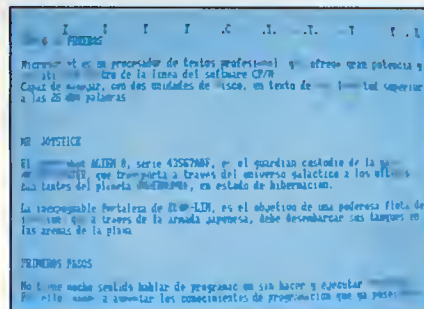
Podemos fijar a nuestro conveniencia casi todos los parámetros de impresión.

## MICROSCRIPT es programable

Queda por comentar, por último dos interesantes características de MICROSCRIPT, que serán bienvenidas sobre todo por usuarios avanzados: la posibilidad de realizar cálculos numéricos y programas empleando los propios comandos de MICROSCRIPT, para automatizar trabajos complejos a que requirieron ser llevados a cabo muchas veces.

La primera convierte a MICROSCRIPT en una primitiva hoja de cálculo: podemos encolumnar números y obtener totales de los mismos vertical y horizontalmente, además de imprimir o guardar en disco los resultados, por supuesto.

La segunda es una clara prueba de la inmensa potencia que posee este programa. Cada comando de MICROSCRIPT puede ser representado de tal modo (*aparte del normal*) que pueden construirse programas muy complejos, con posibilidades interesantísimas, como por ejemplo, aceptar entrada desde teclado en medio de cualquier operación de disco y obrar en consecuencia según las nuevas instrucciones.





# RPA BASE DE DATOS

**El número de programas de bases de datos puede contarse por millares; hay de todos los tipos y concepciones.**



**E**n el caso particular de **Amstrad**, también hay varios. Hemos elegido para Banco de Pruebas uno creada por RPA Systems, que así, a grosso modo, posee dos particularidades interesantes: está escrita en Basic, y no necesita del CP/M para funcionar, es decir, corre bajo Amsdos, sistema operativo nativo de **Amstrad**.

En primer lugar, RPA Systems no entrego manual de instrucciones. Ni falta que hace, fue mi primera reacción al trabajar con él (*luego pude comprobar que existe una opción para ver las instrucciones por pantalla o impresora*), y creo que es acertada. Este programa es la sencillez por antonomasia, es muy difícil equivocarse.

En todo momento, el usuario se ve asistido por mensajes del sistema, absolutamente unívocos, que nos indican, paso a paso, lo que se espera que hagamos o las opciones de que disponemos. El programa está basado en una estructura de menú: existe un menú principal, en el que decidimos lo que deseamos hacer, y luego el programa bifurca hacia la opción elegida.

Pude observar que en este menú, las posibilidades van numeradas, en lugar de acceder a ellas por su inicial.

Bien, supongo que es cuestión de gustos o costumbre, pero uno y otro vez mi dedo se lanzaba a las letras en lugar de a los números, con lo que, o no obtenía nada, o me encontraba metido de lleno en la opción de dar colores a la pantalla cuando pretendía crear un fichero (*si, Color también empieza por C*).

Está muy claro que criticar esto es un tanto

discutible, pero basándome en el hecho de que al resto de las personas que manejaron el programa les ocurrió lo mismo, tal vez hubiera sido más acertado escoger otro sistema. Vedito: en el peor de los casos, mal menor; en la mayoría de las ocasiones, cuestión de gustos. La creación de un fichero se hace de forma completamente interactiva, mediante el uso de ventanas. Usted simplemente mueve el cursor por lo pantalla, fija dos puntos y ya está, tenemos creada una ventana que corresponderá a un campo de lo base de datos, en la cual, una vez etiquetada con algo como «**nombre**», «**cliente número**», o lo que sea, se almacenará la información relativo a ese concepto.

Podemos tener simultáneamente en pantalla hasta 8 ventanas, lo cual no implica que sólo puedan existir 8 campos, esto ha sido previsto.

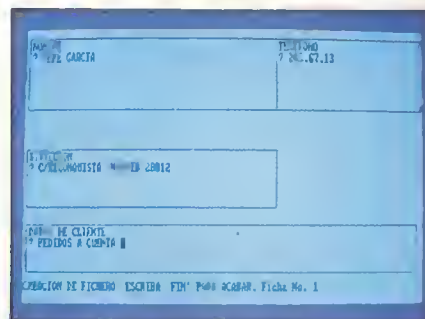
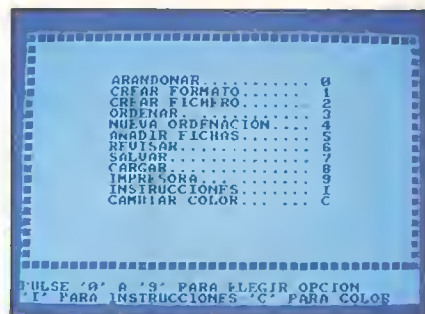
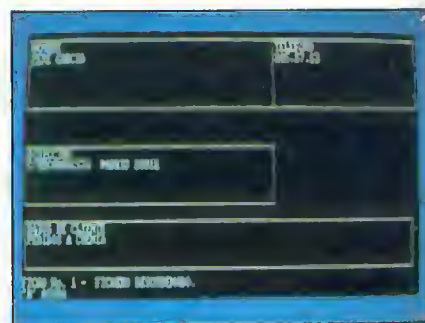
No podemos por menos que felicitar al programador por conseguir lo siempre difícil tarea de crear el formato de una base de datos, de uno formo tan sencilla y elegante.

En el mismo estilo se ubica el tema de dar color a la pantalla diseñada con las ventanas: aparece otra pantalla con una serie de informes acerca de los colores disponibles. Para cambiarlos, se pulsan una serie de teclas y el nuevo set de colores aparece en la pantalla. En definitiva, escogemos visualmente, cómo debe ser, la combinación que más nos acomode. Nudo de cosas como «**Introduzca un número del 0 al 27 para cambiar la tinta...**», o algo igualmente esotérico.

El resto de las opciones son las que cabe esperar de este tipo de programas: ordenación por uno o varios campos, búsqueda selectiva de información por un determinado campo, sacar fichas por impresora, etc., todas igualmente simples de manejar desde el primer momento.

Pudimos observar que el programa se encuentra dividido en multitud de partes en el disco, de forma que cada una está en memoria sólo cuando se la necesita. A pesar de ello, no se observa un retraso «**crispante**» en las operaciones: unos instantes de espera, y nos encontramos donde queríamos estar. Las tareas de ordenación y búsqueda de fichas se realizan también a una velocidad aceptable.

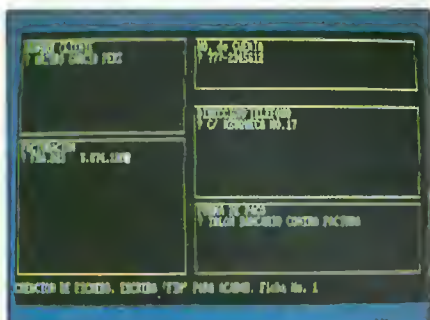
Hablando de ordenación hay que andarse con cuidado con la opción de borrar una fi-



cha: esto se realiza pulsando una tecla y la ficha, sea cual sea, que se encuentre en pantalla, desaparece sin previo aviso.

Esto es lógico hasta cierto punto, ya que supone que el usuario se ha colocado antes en la ficha que quiere borrar, pero puede causar serios disgustos al mínimo despiste. No hubiera estado de más hacer la operación de borrado un poco más dificultoso, o pedir confirmación, o algo que obligará a concentrar la atención sobre lo que estemos haciendo. Una vez borrada, no se puede recuperar.

No conseguimos arrancar el programa en un CPC464 con unidad de disco, sin embargo, funcionó perfectamente en un CPC664 y en el CPC6128.



## FICHA DEL PROGRAMA

CREADO POR:  
**RPA SYSTEMS**  
SOPORTE: **DISCO**  
SISTEMA OPERATIVO: **AMSDOS**  
COMPATIBILIDAD:  
**CPC664/CPC6128**  
P.V.P.: **6.500 Ptas.**



# COMPUTIQUE

*Te da más*

GARANTIA **AMSTRAD**  
ESPAÑA

**66.990 Ptas.**

**Amstrad 464 f.  
verde**

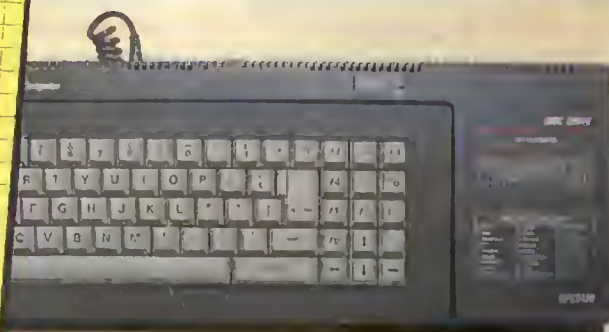
IVA incluido



## Al comprar tu Amstrad te regalamos

- Estuche con ocho programas originales
  - Fruit Machine
  - Procesador texto
  - Almirante Graf
  - Oh Mummy
  - Plaga Galáctica
  - Amsdraw
  - Laberinto Sultan
  - Animal, Vegetal, Mineral
- Joystick Gunshot I
- Un estupendo libro de Basic
- Los cuatro mejores programas:
  - Decathlon
  - Sabrewulf
  - Jet Set Willy
  - Beach-Head
- Guía de referencia del programador
- y además te obsequiamos con un curso de introducción al Basic.

VENTA A PLAZOS HASTA 36 MESES



**Nuevo Amstrad CPC6128: 109.500 ptas. (F. Verde)**

**COMPUTIQUE**

Servimos a tiendas  
Abrimos sábados por la tarde

Embajadores, 90 Tfno. 2270980  
28012 Madrid



# PLACON

**Placon es un programa de compatibilidad completamente adaptado al Plan Contable Español y de lo más potente que hemos visto.**

**Como está dirigido a un sector muy concreto de usuarios de Amstrad, hemos pensado que lo mejor es pasar a la descripción detallada del programa sin más preámbulos.**

**Únicamente una aclaración antes de continuar: Placon requiere para poder funcionar dos unidades de disco. Incluso está prevista su utilización con disco rígido.**

**Las funciones del programa están numeradas y se ha mantenido esta clasificación para ser coherentes con la estructura del mismo.**

## OPCION 10: Mantenimiento del Plan de Cuentas

### A. OBJETO DE LA OPCION

Esta opción permite dar de alto, modificar, dar de bajo y listar las cuentas.

### OPCION 10 — 1: Altas

#### A. OBJETO DE LA OPCION

Introducir cuentas en el maestro, tanto cuentas de agrupación como de detalle.

### OPCION 10 — 2: Modificar

#### A. OBJETO DE LA OPCION

Modificar:

- La descripción de los cuentas.
- La asignación a masa patrimonial.
- La asignación o activo o o pasivo.

### OPCION 10 — 3: Bajas

#### A. OBJETO DE LA OPCION

### OPCION 10 — 4: Listado

#### A. OBJETO DE LA OPCION

Tener un listado del plan de cuentas, tanto por pantalla, como por impresora.

## OPCION 11: Mantenimiento del Fichero de Masas Patrimoniales

### A. OBJETO DE LA OPCION

Las masas patrimoniales se usan para agrupar cuentas de mayor, según el grado de liquidez, o imprimir el balance de situación.

## OPCION 12: Mantenimiento del Fichero de Conceptos

### A. OBJETO DE LA OPCION

El fichero de conceptos contiene 45 descripciones de 20 caracteres cada uno que utilizaremos para facilitar y ahorrar espacio al introducir los asientos.

## OPCION 20: Anotación de Asientos

### A. OBJETO DE LA OPCION

Anotar los movimientos de la contabilidad.

## OPCION 30: Modificación de Apuntes

### A. OBJETO DE LA OPCION

Esta opción permite la modificación de cual-

quier campo, de cualquier asiento, de cualquier asiento ya actualizado.

## OPCION 31: Incorporación de Asientos Externos

### A. OBJETO DE LA OPCION

Esta opción lee ficheros realizados por otros aplicaciones (Námina, Facturación —ALFA—, e incluso Contabilidad General y los incorpora a la contabilidad en curso.

El proceso es parecido al de introducción de asientos. La diferencia es que, en vez de leer el teclado, se leen los datos del fichero grabado por otra aplicación.

Los asientos se incorporan al fichero auxiliar de entrada de apuntes. Puede modificarlos con la opción de anotación de asientos exactamente igual que si lo hubiera tecleado. Hasta que no la actualice no se consideran en los listados.

## OPCION 40: Fin de Apertura

### A. OBJETO DE LA OPCION

Actualizar el campo del saldo inicial de ejercicio.

## OPCION 41: Fin de Mes

### A. OBJETO DE LA OPCION

Actualizar los campos sumas debe mes anterior y sumas haber mes anterior.

## OPCION 42: Fin de Período

### A. OBJETO DE LA OPCION

Es actualizar los campos de sumas del debe y sumas del haber del período anterior e inicializar el fichero de asientos.

No es prudente realizar este proceso si na-

hay dos copias del disco de datos (se guardarán como históricos).

## OPCION 43: Regularización de Cuentas de Gestión

### A. OBJETO DE LA OPCION

Generar un asiento cuyos apuntes saldan las cuentas de gestión.

## OPCION 44: Fin de Ejercicio

### A. OBJETO DE LA OPCION

Poner a cero los acumulados del debe y del haber de todos las cuentas y generar el asiento de apertura (que queda en el fichero auxiliar de entrada de apuntes) también inicializa el fichero de asientos.

## OPCION 50: Listado del Diario

### A. OBJETO DE LA OPCION

Listar el diario.

## OPCION 51: Listado del Mayor

### A. OBJETO DE LA OPCION

Listar el mayor.

## OPCION 52: Balance de Sumas y Saldos

### A. OBJETO DE LA OPCION

Listar un balance de sumas y saldos.

## OPCION 53: Balance de Situación

### A. OBJETO DE LA OPCION

Obtener el balance.

## OPCION 54: Cuenta de Explotación

### A. OBJETO DE LA OPCION

Obtener el informe.

## OPCION 55: Explotación Analítica

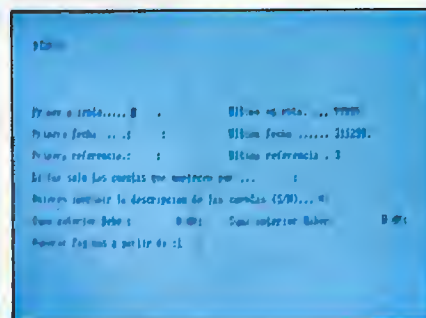
### A. OBJETO DE LA OPCION

Obtener el informe.

## OPCION 60: Previsión de Cobros y Pagos

### A. OBJETO DE LA OPCION

Mantenimiento del fichero de previsiones y listado clasificado de previsiones de cobros y pagos.





# CONTABILIDAD

**El programa de contabilidad, adaptado al Plan Contable Español, destaca fundamentalmente por dos cosas: 1. Es increíblemente sencillo de usar. 2. Puede emplearse con una o dos unidades de disco. Para analizarlo, seguiremos manteniendo el mismo enfoque adoptado con el Placon, sin pretender por ello entrar en comparaciones, siempre odiosas y en este caso más, porque cada uno de los programas cumple su función dentro del hueco que pretende llenar. Sin más preámbulos, vamos a pasar a describir detalladamente el programa:**



Las funciones del programa de Contabilidad están divididas en seis apartados, que son:

1. ASIENTOS CONTABLES.
2. PLAN CONTABLE.
3. LISTADO DE CUENTAS DE MAYOR.
4. BALANCE DE SUMAS Y SALDOS.
5. CUENTA DE RESULTADOS.
6. CIERRE PERIODICO.

## Asientos contables

En esta opción están recogidos todos los procesos a realizar con las asientos de la Contabilidad.

Al entrar en este capítulo nos aparecerá en la pantalla, una menú, con las siguientes apartados:

- ENTRADA DE ASIENTOS.
- CONSULTA DE ASIENTOS.
- LISTADO DEL LIBRO DIARIO.

### ENTRADA DE ASIENTOS

Consiste en la grabación de las partidas contables, como información de base para la elaboración de la totalidad de documentos contables que se pueden obtener con este programa.

### CONSULTAS DE ASIENTOS

Listada en pantalla de las asientos, discriminadas entre dos fechas.

Se pedirá una fecha inicial, y otra final, encuadrando así las asientos a consultar en un periodo de tiempo.

### LISTADO DEL LIBRO DIARIO

Al igual que el anterior proceso, pero esta vez sobre el papel, el listado del libro diario imprimirá las asientos limitadas por dos fechas dadas.

## Plan contable

Este apartado recoge todas las funciones a realizar con el Plan de Cuentas.



Tenemos el siguiente menú:

- ALTA DE CUENTAS.
- MODIFICACION DE CUENTAS.
- CONSULTA DE CUENTAS.
- LISTADO DEL PLAN CONTABLE.
- CLASIFICACION DE CUENTAS.

Cama ya dijimos en el apartado de Entrada de Asientos, el primer proceso a realizar con el programa, será la definición de las cuentas, es decir...

### ALTA DE CUENTAS

Permite la grabación en el archivo de todas las cuentas que conforman el Plan.

### MODIFICACION DE CUENTAS

Cuando nos percatamos de que el título de una cuenta no está bien escrita, a simplemente queremos cambiar su denominación a otra más exacta, recurriremos a este proceso, que es casi idéntico al anterior. Se requiere el número de cuenta a modificar, para que el ordenador la busque en el archivo.

### CONSULTA DE CUENTAS

Listada por pantalla de parte a la totalidad del Plan Contable, así como la suma de DEBE, HABER y SALDO.

### LISTADO DEL PLAN CONTABLE

Para tener un listado impreso de las cuentas, pero en papel, y sin los saldos, únicamente el número de cuenta y título.

Nos pedirá fecha de edición, para salir impresa en el listado.

### CLASIFICACION DE CUENTAS

Este proceso debe realizarse forzadamente siempre que se produzcan altas en el Plan Can-

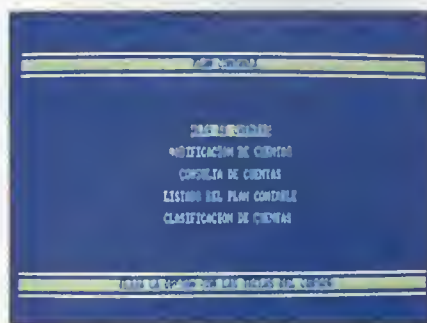
table. De no realizarse, las cuentas dadas de alta después de la última clasificación, no podrán ser usadas, como si no existieran.

## Listado de cuentas de mayor

Para obtener el listado de asientos de una determinada cuenta y en determinada fecha, utilizaremos este proceso, en dos opciones diferentes:

- LISTADO POR PANTALLA.
- LISTADO POR IMPRESORA.

Cama su propio nombre indica, los dos apartados realizan la misma función, con la salvedad de que el primero muestra el resultado en pantalla (para ir confeccionando el Libro-Mayor).



## Balance de sumas y saldos

Para la obtención de listado del Balance de Sumas y Saldos, en la impresora.

En este caso no hay ninguna opción a tomar tan sólo se requiere la «Fecha de Edición».





# DECISION MAKER

*Es un programa de ayuda a la toma de decisiones, esto es, el Amstrad, asistido en este caso por usted mediante una serie de datos, es capaz de indicarle el camino más lógico (¡ojo!, no el más correcto necesariamente) para cumplir un objetivo basado en determinadas premisas.*

*El Programa está bien documentado, y lleva en el disco un programa tutor para el no iniciado en la teoría del Análisis de Decisión.*



Quando el programa esté en marcha, mostrará el menú de opciones iniciales:

Comience un Nuevo Arbol  
Cargue un Arbol Anterior  
Finalice este programa

## Construcción de un Arbol

Cuando se elige la opción de Comenzar un Nuevo Arbol, la pantalla se queda en blanco y el programa trazará el primer nodo decisivo en la parte izquierda. En la parte superior se ven los mensajes «Evento Anterior; Decisión Inicial».

En la parte inferior de la pantalla hay un mensaje que solicita que introduzcas la cantidad de acciones diferentes que van a emerger de este nodo.

Las preguntas se repiten sucesivamente para cada una de las ramas. Cuando hayas definido todas las ramas, aparece un menú en la parte inferior izquierda de la pantalla. El primer menú ofrece cuatro opciones:

**Siga:** El ordenador pasará al siguiente nodo, buscando la información requerida para completar el árbol.

**Todo:** Visualiza toda la imagen del árbol completado hasta ese momento.

**Copie:** Si tienes una impresora Amstrad DMPI, esta opción hará que se imprima en papel una copia de lo que hay en pantalla, y luego continúe con el programa.

## Búsqueda de la solución

Cuando el árbol ha sido completamente definido, al elegir en cualquiera de los menús la opción hará que el ordenador proceda a calcular el **valor medio esperado** en el modo inicial del árbol y que regrese al menú principal, y se te ofrecerán las siguientes alternativas:

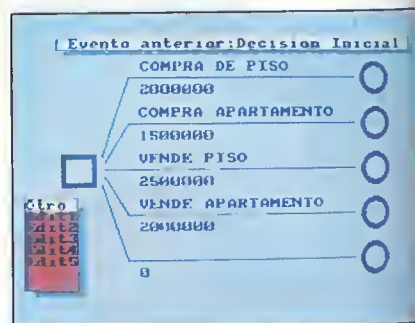
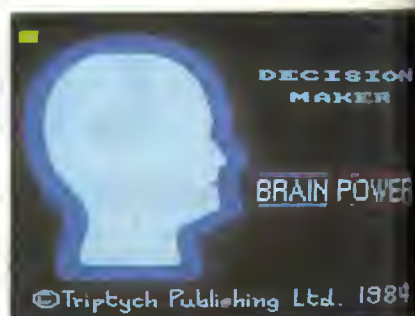
Comenzar un Nuevo Arbol  
Cargar un Arbol Anterior  
Mostrar Decisión Inicial y Revisar  
Calcular Perfil de Riesgo  
Grabar este Arbol  
Finalizar este Programa

Puedes revisar cada nodo en detalle si lo deseas, eligiendo la opción de Revisar y cuando ya estés preparado, puedes regresar al menú principal mediante la opción de Seguir. Si consideras que debes hacer alteraciones a los datos del árbol, puedes hacerlo mediante las opciones de Revisar y Juntar, pero en este caso la opción Seguir sólo te volverá al menú principal si el árbol queda completamente definido, y en ese caso además volverá a calcular el valor medio esperado.

Cuando estés de nuevo ante el menú principal, puedes elegir la opción de Calcular el Perfil de Riesgo. Las probabilidades y los costes o valores para todos los posibles resultados de los nodos aleatorios, aparecerán entonces en pantalla. Puedes regresar al menú principal o sacar copia de este **Perfil de Riesgo** mediante la impresora, si dispones de ella.

## Valor de la información de Muestreo/Sondeo

Finalmente, puede obtenerse del árbol una información adicional. Usa las opciones ya



mencionadas para mostrar otra vez todo el árbol. Si ahora mueves el cursor hasta cualquier nodo aleatorio y eliges la opción de Revisar, el nodo aparecerá con los detalles completos tal y como es habitual. Siempre y cuando haya sido calculado el valor medio esperado, aparecerá una opción adicional en el menú denominado **EVSI** (*valor esperado según información de «sondeo»*). Eligiendo esta opción hará que se calcule el valor esperada según la nueva información adicional correspondiente a ese nuevo nodo aleatorio. El ordenador te hará dos preguntas:

### ¿Fiabilidad del sondeo?

— Teclée la fiabilidad que tú estimes como un número de 0 a 1. El valor 1 representará la **información perfecta**, y si lo das lo que obtiene realmente es el valor **EVPI** (*valor esperado según información perfecta*).

### ¿Valor de la opción cero?

— ¿Cuál es el coste de evitar este riesgo? i.e. si se predice un resultado con un valor medio esperado negativo, ¿cuánto costará hacer algo más, o no hacer nada en lugar de eso? Típicamente, ante esta pregunta se puede contestar que **0**.

## FICHA DEL PROGRAMA DECISION MAKER

Comercializa: Indescomp.  
Soporte: Disco.  
Sistema operativo: CPIM 2.2  
Compatible con: CPC464, CPC664, CPC6128.  
Precio: 5.500 ptas.



Defiende tu espacio aéreo

**EL Nº 1**  
**ahora en**  
**AMSTRAD**



Simulación de vuelo tridimensional de combate aéreo y ataque al suelo.

Cinco niveles, quince escenarios y capacidad de juego estratégico.

Estás en la cabina del caza que sería el sueño de cualquier piloto, pero desde luego eres un mal sueño para el pobre tipo que tienes delante, confiado en una misión sin problemas. Calientale la tobera con tus láser y apartate mientras estalla en una bola de fuego. Rápidamente ponte en picado para caer sobre los blindados

enemigos, como la peste entre los cerdos. SKYFOX es el juego que más rápidamente se está vendiendo en toda la historia de Electronic ARTS.

Posee la más asombrosa animación de alta velocidad que hayas visto en tu ordenador. Ahora puede ser tuyo totalmente traducido al castellano.

# SKYFOX™

EN CASTELLANO



DRO SOFT

P.V.P. 2.500 ptas.

ELECTRONIC ARTS

**CARACTERÍSTICAS:** NACIONALIDAD: Federación galáctica. **FABRICANTE:** TOBEY ASTRONAUTICS. **TIPO:** Caza interceptor multipropósito. **PROPULSION AUXILIAR:** Un generador antigravitatorio a 66 MKI. **TRIPULACION:** Un humano. **ARMAMENTO:** Dos cañones láser de fuego continuo de 70 kilojulios. 10 toneladas de empuje. 5 misiles rastreadores de calor tipo PHOENIX. 5 misiles guiados por radar tipo TYPHOON. **DEFENSA:** — 2 unidades deflectoras WCRC. **AYUDAS ELECTRONICAS:** Radar SCANNER de largo y corto alcance conectable al piloto automático. **VELOCIDAD EN ATMOSFERA:** — 3.000 MPH (Mach IV a 35.000 pies).

Editado por DRO SOFT Fundadores, 3 - 28028 Madrid  
Tel.: 255 45 00/09



# Los Super DIEZ

**En la ya prolongada vida del software de juegos para Amstrad, hemos tenido la oportunidad de ver programas de todos tipos y colores; unos mejores, otros de menos calidad, unos originales, otros siguiendo una temática ya iniciada por otras marcas.**

**Ante esta verdadera invasión de productos, Superdiez pretende dar una panorámica de los diez juegos en la historia del Amstrad.**

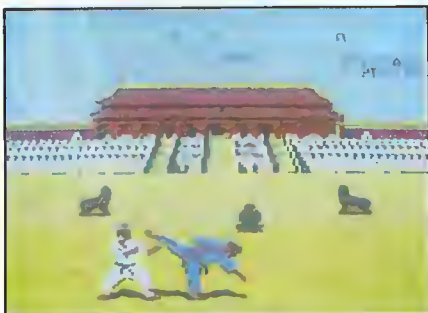
## 1 EXPLOIDING FIST

SOLAMENTE 464.

MELBOURNE HOUSE, siguiendo la clara tendencia deportiva de muchas firmas de software, ha creado una magnífica versión de kárate para ordenador personal. Tarea bastante difícil, debido a que la belleza de esta sublime disciplina reside en la variedad de golpes y pasiones que las luchadores pueden adaptar durante un combate.

El juego tiene como base una pelea entre dos luchadores. Permite las distintas opciones, de jugar contra el ordenador, o jugar dos personas entre sí.

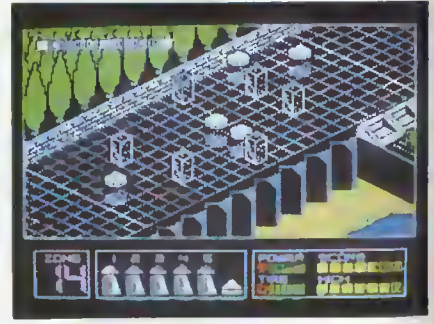
La verdaderamente asombrosa del **EXPLOIDING FIST**, es el movimiento de las karatecas. Cualquier aficionado a asiduo practicante de las artes marciales, observará que las técnicas utilizadas por los luchadores son perfectas, la ejecución de cada golpe es un reflejo auténtico de la realidad y el dinamismo y ritmo del combate con ataques, sucesivas defensas e intercambio de golpes no puede ser más acertada.



El jugador dispone de 18 golpes y técnicas distintas que puede utilizar en la lucha.

Todas ellas estudiadas minuciosamente y con una animación perfecta.

Las niveles que contempla el juego van desde principiante hasta décimo dan. Un combate se gana cuando se consiguen dos puntos, cada golpe ejecutado con acierto sobre nuestra adversaria, incrementa el número de puntos, una forma adictiva para conseguir nuestra récord; bien sea por los puntos alcanzados, o por el dan conseguido.



## 2 HIGHWAY ENCOUNTER

Guardando estricta formación, los robots deben avanzar por la pista; en cabeza el cabo robotará y detrás en columna el resto de la trapa.

Todo el desgaste y el peso de la lucha lo lleva el cabo; éste combatirá hasta la muerte. Mientras tanto, la trapa viaja detrás, cubierta del fuego enemigo por el cabo de formación, esperando el momento de entrar en acción cuando éste caiga.

Hasta ese momento, sólo se acupan de empujar la tetrabamba que deben depositar en la base.

Las antorchas letales, prismas cristalinos, columnas vulnerables a nuestros disparos, prismas pétreos, estrellas de la muerte, robots boca y platillos flatadores, son algunos de los peligros y objetos que nos pueden aniquilar, o de los cuales podemos sacar algún partido en nuestro beneficio.

Llegar hasta la base Alfa no es cosa fácil, en el camino hacia ella muchas de nuestros androides caerán, e incluso nos quedaremos a la puerta de la misma, sin poder alcanzar nuestro objetivo.

El decorado de las distintas pantallas, está realizado con un cuidado exquisito, transportándonos a una era mecánica, donde los robots, y distintos seres mecánicos, son protagonistas principales.

Descubrir nuevas pantallas, según avanzamos hacia la fatídica zona, es una completa gaza.

Un digno exponente de la nueva generación de software.

Compatible: CPC/464, CPC/664 y CPC/6128.



## 3 KNIGHT LORE

SABREMAN está poseído por un hechizo que le transforma en hombre lobo, con la salida de la luna.

Nos encontramos en el castillo del mago Melkior, donde vagando por sus intrincadas salas y estancias, debemos ir recogiendo los ingredientes necesarios para componer la pócima que acabará con el hechizo maldito.

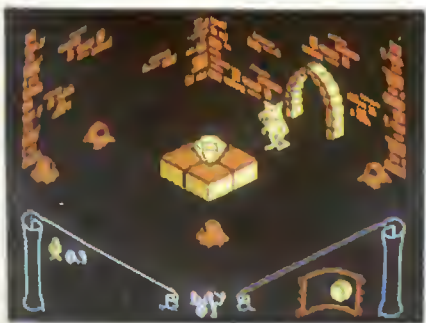
Estos deben ser vertidos en la gigantesca caldera de Melkior en un orden determinado, de forma que si nos acercamos a la olla mágica con el ingrediente adecuado, la nube de estrellas que emana de la caldera se convertirá en centelleante. En caso de no llevar el ingrediente preciso, seremos atacados por la masa de estrellas obligándonos a abandonar la estancia a toda prisa.

Distribuidos por las distintas salas, se encuentran diversos objetos que son imprescindibles para superar los obstáculos que surgen a nuestro paso.

Colocándolos estratégicamente y subiéndonos en ellos podremos escalar muros demasiado altos para nuestra estatura.

El castillo está formado por un intrincado laberinto de 128 salas, plagadas de obstáculos y los más inesperados peligros, en algunas de las cuales se encuentran repartidos los objetos que necesitamos para lograr el éxito.

Sacar a SABREMAN del hechizo maléfico que le posee es una tarea que sólo tú puedes lograr, suerte.



Compatible: CPC/464, CPC/664 y CPC/6128.



## 4 STARSTRIKE

Compatible: CPC/464, CPC/664 y CPC/6128.

La Estrella de la Muerte domina bajo el yugo del miedo a toda la galaxia, aplastando cualquier insubordinación solamente con su proximidad; su fuerza y poderío son tan descomunales que cualquier planeta que la divisa teme por su existencia.

El ambiente de rebelión se extiende por todo el imperio y una fuerza de combate se dispone a intentar el asalto final a la poderosa estrella.

Estamos a los mandos de nuestro caza intergaláctico.

El primer escollo son las naves enemigas que aparecen en oleadas y disparan sobre nosotros. Podemos detener los disparos con nuestro fuego y debemos destruir el mayor número posible de naves.

Superada esta primera prueba, pasamos al planeta, en el que debemos hacer fuego sobre zonas de energía radiante.

En sucesivas fases, nos encontraremos con columnas de energía que crecen delante de nuestra vida disparando láseres cuando su altura ha llegado al máximo.

Por fin entramos en la Estrella de la Muerte, estamos en uno de los corredores, obstáculos a distintas alturas interceptan nuestro paso por ellos, los láseres intentan alcanzarnos mientras destruimos las escotillas de fuego.

Si salimos del túnel, llegaremos al sistema central de reactores, un disparo certero es suficiente para destruirlo, si lo conseguimos habremos salvado a la rebelión de las fuerzas imperiales.

Starstrike, es simplemente fantástico, los efectos en tres dimensiones son los mejores logrados en la historia, la acción es trepidante, la velocidad se siente.

Un juego para los amantes del Joystick y la acción.



## 5 DUN DARACH

Bajo el sortilegio de SKAR, LOEG ha sido llevado a la misteriosa ciudad amurallada de DUN DARACH y encerrado en cuerpo y alma en el castillo de ésta.

Su fiel amigo, sin dudarlo un momento, monta un caballo y parte en busca de la misteriosa DUN DARACH, el camino es largo y penoso, pero al fin CUCHULAIN atraviesa los tétricos muros y camuflado entre los habitantes comienza la búsqueda.

El programa está ambientado en una ciudad celta, con personajes muy bien caracterizados; los hombres con largas melenas y botas, y las mujeres con faldas cortas y una belleza fuera de lo común en este tipo de juegos.

Cabe destacar el maravilloso movimiento con el que se ha dotado a los personajes; es fascinante observar cómo camina nuestro protagonista con sus melenas acompañando el ritmo de sus pasos en una conjunción muy realista y bella.

CUCHULAIN puede robar, comerciar, cambiar objetos por información, vender mercancías y otras tareas que se descubren a lo largo del juego.

GARGOYLE GAMES ha realizado una aventura para aquellos amantes de los juegos de larga duración, en los que es importante el uso de mapas confeccionados por los propios jugadores y con la innovación de que hemos de utilizar a fondo nuestro sentido comercial y especulativo.



Compatible: CPC/464, CPC/664 y CPC/6128.



## 6 YIE AR KUNG-FU

**Las artes marciales siguen siendo un caudal inagotable, para la creación de juegos de ordenador. En esta ocasión, se trata de una adaptación de un popular juego de las máquinas de los bares, el Yie Ar Kung-Fu.**

El primer adversario es el terrible Buchu, un peso pesado, con el extraño poder de volar, unido o una desmesurada fuerza.

Superado éste, nos enfrentamos a Star, bellísima mujer, dominadora del arte de lanzar las estrellas mortíferas.

El siguiente es Nuncho, el luchador diestro en el manejo de los Nunchacus.

A continuación le toca el turno de combate a Pole, el hombre cuyo arte de lucho se basa en la coña de bambú.

Club, el próximo en la lista, no tiene nada que ver con los anteriores.

Sus armas son de lo más efectivas, utiliza un escudo hasta los pies y uno porra pesada, cuyos golpes son demoledores.

Sword, el maestro de la espada curva, que ejecuta los soblazos con una maestría impecable.

Tonfun, el tailandés capaz de eliminar a cualquier diestro luchador, con sus efectivas Tonfas.

Si Bruce Luis consigue superar a esta serie de expertos adversarios, aún le queda la prueba de fuego: Blues.

Un juego de combate, en el cual nuestro omigo ejecuta 10 técnicas distintas, con un repertorio de movimientos más que aceptable.



Compatible: CPC/464, CPC/664 y CPC/6128.



## 7 SABRE WULF

Compatible: CPC/464, CPC/664 y CPC/6128.

«La senda es larga y tortuosa. El peligro amenaza en la guarida del lobo. Por el camino de lo selvo, al encuentro de tu destino, un omuleto debes encontrar. Fue roto en cuatro partes, y escondido en claros de la jungla. El guardián nunca te permitirá entrar sin él.

Esta leyenda, encontrada en una antigua losa, es leída por nuestro explorador, el cuol, cautivado por el misterioso secreto escondido en la cueva del hechicero, no puede resistir la tentación y comienza la búsqueda. Rodeado por la exuberante vegetación, Sabreman elige una senda, por la que encamina sus pasos hacia un destino desconocido. Cocoteros, palmeros y demás plantas tropicales, nos acompañan en nuestro recorrido, escondiendo los peligros que nos acechan. Ocultos en los claros de lo jungla, se encuentran las cuatro partes del mágico talismán, sin el cual no podemos eludir al guardián de la cueva del hechicero.

Una vez recogidos los cuatro, podemos decir que la aventura está en nuestras manos, sólo nos queda dirigirnos a la cueva y desvelar su secreto.

El escenario que recorre el intrépido explorador, está formado por 256 pantallas que hemos de recorrer para encontrar los preciados trozos del talismán.

La estructura laberíntica de los caminos de la jungla y la distribución de los claves, nos obliga a elaborar un mapa de los distintos pantallas, para saber dónde nos encontramos y la ruta a seguir.

Sin éste, vagaríamos por las mismas sendas continuamente, dejando zonas inexploradas que nos impedirían acabar el juego.

## 8 FIGHTING WARRIOR

Lo oventura está ambientada en el Egipto de las pirámides, con una variedad de personajes y decorados que le don un atractivo extraordinario.

Nuestro personaje, en esta ocasión es el joven Simuel, el último campeón de su región de lucha con vara de caña.

Cuando el sol aparece por el horizonte, bebe la pócima que le dará la fuerza necesaria para vencer o morir y se dirige hacia el templo.

Uno tras otro, los más demoníacos seres intentan cortar su camino hacia el pórtico de entrada.

Seres con cuerpo humano y cabeza de león, cabezas de hieno, y demás animales sagrados.

Además de los oporiciones de humanoides, terribles dragones alados, aparecen de lo nada y le atacan con sus terribles garras.



La lucha continuará uno a uno hasta que Simuel, dotado de la extraña fuerza de la pócima, llegue hasta la princesa, o muera en el empeño, tal es la importancia de su sagrado misión.

Llegar hasta el templo no es tarea fácil, en cada combate, los golpes recibidos, nos restan energía vital; por si todo esto fuera poco, flechas lanzadas desde larga distancia nos hieren restandonos energía.

Todos conocemos la gran pericia de los autores del Exploding Fist, en la creación de gráficos y en los hiperrealista efectos de animación que son capaces de crear.

Pues en este juego no se quedan atrás, la secuencia de gráficos que generan los distintos golpes, están ejecutadas con un gran sentido del movimiento.



Compatible: CPC/464, CPC/664 y CPC/6128.





## 9 ROCKY HORROR SHOW

SOLAMENTE CPC 664.

**El malvado Dr. Frank, ha petrificado al joven Brad, la única forma de salvarle, es que Janet encuentre la máquina que realiza la despetrificación.**

La Desmedusa, que así se llama la máquina, está fraccionada en componentes los cuales se encuentran repartidos por toda la casa.

Janet solamente tiene que encontrarlos y recomponer Desmedusa, para salvar a Brad.

Misión difícil, porque solamente dispone de 100 minutos y por el hecho de que por la mansión circulan extraños personajes que pueden acabar con ella.

El Rocky Horror Show, es un juego emocionante, en el que el conocimiento de las distintas habitaciones y salas de la mansión es imprescindible para conseguir el éxito.

Descubrir las habitaciones secretas y utilizar el ascensor a nuestra conveniencia son armas claves.

También es importante saber los personajes que son mortales para nosotros y los que solamente rebajan nuestra energía.

El decorado de las distintas salas es muy acertado, con unos gráficos tipo cómic muy cuidados.

Nuestro personaje, puede utilizar palancas y botones para abrir puertas, desactivar trampas, poner en marcha la calefacción y demás cosas que sólo se aprenden con un poco de práctica.

Una aventura con buenos gráficos y de duración limitada, con lo cual, evitamos el tedio de este tipo de aventuras interminables en las que nos dedicamos a vagar por parajes desconocidos en busca de no sé qué objetos y con un objetivo que tampoco conocemos, lo que después de varias horas de búsqueda nos induce a arrinconar el juego y dedicarnos a otra cosa.



# 10

## FRANK BRUNO'S BOXING

SOLAMENTE 464.

**Elite crea un programa dedicado al deporte de las doce cuerdas, en el cual los más terribles luchadores, intentarán hacernos besar la lona.**

**Para conseguir incluir un número elevado de contrincantes, se ha encontrado una solución muy acertada.**

El programa utiliza una cara de la cinta para las rutinas de movimiento, golpes, marcadores, tiempo, etc., y la otra para cargar los distintos adversarios.

Con este sistema de multicarga, revolucionario en los juegos de cinta, se consiguen unos resultados francamente buenos, pudiendo al finalizar una fase de juego con el código obtenido, cargar la fase siguiente, que contiene gráficos distintos y aumenta la aventura en una longitud considerable.



Los contrarios que nos separan de la corona mundial son: CANADIAN CRUSHER, FLING LONG CHOP, ANDRA PUNCHEREDOV, TRIBAL TROUBLE, FRENCHIE, RAVIOOLI, MAFIOSI, ANTIPODEAN ANDY y PETER PERFECT.

**FRANK BRUNO'S BOXING**, es un juego ante todo superadictivo, vencer a un púgil para ver cómo es el siguiente y cómo pelea es una verdadera obsesión, los distintos contrincantes hacen que el programa adquiera verdadero interés.

La gama de golpes y movimientos de FRANK es muy amplia; guardia arriba, guardia abajo, opercut, directos de izquierda y derecha a la cara y estómago, esquiva a la derecha, esquiva a la izquierda y agacharse.

Todo esto aderezado con un movimiento veloz y unos buenos gráficos, un programa de horas y horas de entretenimiento.





# UN DÍA EN LA VIDA DE LINCOLN FREUD ONERR

**D**esperto despacio, como a desgana. Desde algún lejano y oculto rincón de su mente, los musicales apremios de su programa simbiótico le arrancaron, suave pero implacablemente, de las garras de algodón del sueño. Podía sentir cómo iba tomando conciencia de su cuerpo músculo a músculo, nervio a nervio; Progíe le informaba con la alegre precisión de siempre que todo marchaba bien:

- sistema nervioso central: operativo.
- Sistema cardiovascular: operativo.
- Total check: afirmativo.

Onerr, tranquilizado, dejó transcurrir un poco de tiempo más en esa deliciosa lasitud que sigue al despertar de un sueño profundo, ignorando las tentativas de Progíe para hacerle levantar.

De pronto, recordó que hoy podía ser uno de los Días Importantes. ¿No era mañana?

—Progíe, ¿qué día es hoy?

—Searching System Date.

—File Open:

Martes, 25 de julio del 2150 de la Simbioera.

Región occidental (Tierra).

Gobernador: Sistema Lincoln.  
Moderador mental: Sistema Freud.  
File close.—

—¡Maldita sea! —estalló Onerr—.

—¿Cuántas veces tengo que decirte que no me des información innecesaria? Quiero saber si hoy es uno de los Días, no el día del año ni el lugar en que vivo—.

Progíe zumbó desconcertado durante unos nanosegundos, mientras establecía contacto con Madre, pero inmediatamente respondió con su habitual parsimonia: —Tu pregunta ha sido contestada en base a la información suministrada. Precisa el ámbito de la respuesta de forma lógica—.

Se habría encogido de hombros si los hubiera tenido, pero en la última Epoca de Mutación se colocó un par de brazos extra precisamente ahí; con el consentimiento de Padre, por supuesto.

Con un suspiro de resignación, Onerr pensó que un programa simbiótico no sólo tenía ventajas: había que saber preguntarles exactamente lo que uno quería saber. Es cierto que Progíe le había acompañado desde el mismo momento de su nacimiento, cuando le fue implantado en su médula espinal, y que gracias a él nunca se había sentido solo, y tenía acceso a la Red de todo el planeta, pero a veces, bien, a veces le gustaría poder

desconectarse, tener algo de... ¿de intimidad? ¿De dónde habría sacado esa palabra? Sí, no estaría mal desconectarse un ratito y...

—Atención, para Sistema Freud de Simbioprograma Onerr. Solicito canal de comunicación prioridad 5. HABLE.

—Sujeto Onerr. Donador Emocional clase Alfa en situación de alerta azul. Transmíto mentograma.

RECIBIDO.

PROCESANDO.

IF ALERTA (tipo) < = AZUL AND  
HOMBRE (CLASE) = ALFA THEN  
BORRAR (MENTEGRAMA, 80%)  
SUSTITUIR (DIA)

ELSE

DOLOR (PLEXO, 5)

ENDIF.

REGLA 3/A AFIRMATIVA. BORRE.

Borrado. Sustitución activa. Gracias, Madre.

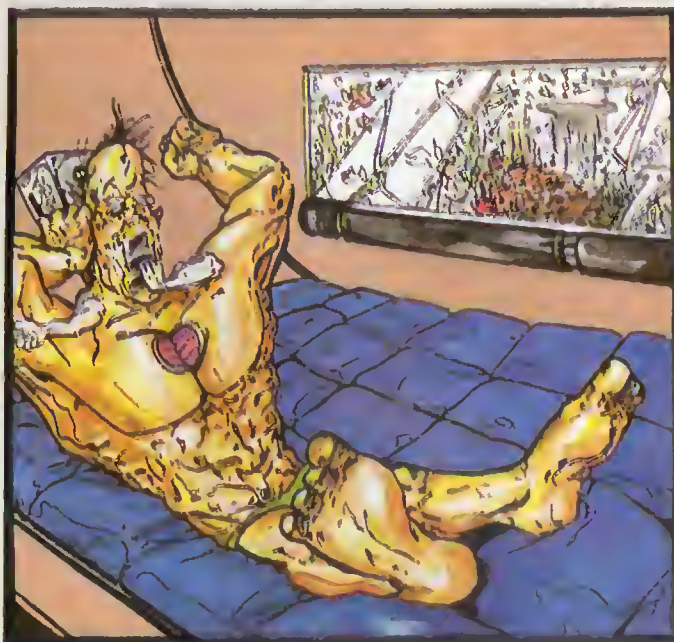
TERMINADO.

... era una mañana realmente preciosa. Menos mal que se había levantado hacía por lo menos tres horas para poder admirarla a placer. Además, Progíe le había confirmado al despertar que hoy era el Día del Amor; Madre estaría esperándole impaciente. Al fin y al cabo, él, Onerr, era un Donador clase Alfa, recordó con orgullo.

—Progíe, por favor, llévame al Terminal. Quiero unirme a Madre—.

Aparecieron en el interior de un pasillo ovalado de color verde esmeralda.

Mientras se reponía de la ligera náusea que siempre le provocaba la Transición, Onerr se preguntó, en uno de sus escasos momentos de curiosidad, si el Terminal se





encontró en el mismo edificio donde él vivió en el período octavo, o estubo en otra edificio o en otro continente; bueno, realmente no impartió demostración, comentó en voz alta. El suelo de bioplast que le transportaba o lo largo del pasillo le dio la razón fervorosamente.

—Realmente, no impartía demostración.

—Remacho el suelo de nuevo.

—Progie, este suelo no parece muy despierto. Se limita a repetir lo que yo digo.

—Hoy mucho tiempo que me transportaba a un Donador Alfa hasta Madre. Estoy debilitado. Todo esta sección de Padre está debilitado. Por eso he decidido venir aquí.

—¿Ya he decidido venir aquí? Hum..., sí, supongo que sí. Parece simpática este suelo.

Y, ¿qué tal la llevo la pared?

—sanaría Onerr, cruzando sus cuatro brazos detrás de la espalda, con el oído más inerte que pueda.

—La pared me llevo a ti. ¿Que podría llevar una pared? Esta simplemente poro...

—¡Vete o lo muerdo!

—¿Ahora misma?

—¡No! Orden onuloso, ¿me oyes?, orden onuloso. Sigamos hasta Madre—.

A Onerr todavía le entorpecían sudores fríos cuando recordaba DONDE había oído cuando Prágie interpretó sus palabras literalmente la última vez; en realidad, le aterraba recordarla.

De pronto, el suelo pasa de verde a rojo escarlata, y con un respingo tremendo, envía a Onerr y a su simbiopragma de cabeza contra la pared; esto trató de envolverlos en sus tibias y polvitos pliegues, besándoles entre tanta oposición. Prágie dio la voz de alarma inmediatamente:

—Onerr, ¡cástrate!, estás perdiendo

histórico o toda esta sección de Padre.

Damna tus emociones. Lo poro cree que es el momento de la Unión y el suelo está celoso. Cómolos; eres un Alfa. Ellas se distinguen bien entre tipos de emociones—. Onerr lo hizo con facilidad. La pared se calmó rápidamente y el suelo recuperó su tranquilidad al calar verde esmeralda. La puerta de Terminol se hizo visible y... entró al apartamento de Madre.

Ero difícil de sentirse impresionado y empujé al entrar en Terminol.

Onerr se encontró de pie en una inmensa sala, o al menos, lo percibió trapezoidal de lo estoncio lo hizo aparecer inmensa, materialmente repleto de formas geométricas y difusas, palpitantes y translúcidas: las órganos esclavas de Madre.

Sintió en su mente y en cada célula de su cuerpo el poder de un dulce cántico de bienvenida, amarga pero llena de fuerza y de urgencia de él. Madre estaba impaciente, Padre estaba impaciente, Prágie estaba impaciente, él... ¿Na estaba impaciente?

Un vago recuerda, nebulosa y desagradable, trataba de abrirse paso a empujones en su conciencia; ¿Cómo era? ¿Cómo demanías era? Ah, sí...

—Atención, poro sistema Freud de Simbiopragma...

LO HE DETECTADO, ESTUPIDO. SUJETO ONERR EN ALERTA AMBAR.

Estoy recordando, estoy intentando recordar con todas sus fuerzas lo que pasó en otras ocasiones.

NO DEBE HACERLO, NO LO HARA.

BORRADO TOTAL.

IMPRIMACIÓN SEXUAL FUERZA 5.

Madre, es peligrosa. El sujeto es un Danador altamente inestable. Puede verse doñodo. Ya pueda ser dañada.

HAZLO. ¡AHORA!

Barrada total: afirmativa. Imprimitiva sexual: operativa. TERMINADO.

... que otroctiva es Madre, pensó Onerr, y su simbio le dio la razón.

—Prágie, miro qué forma he tomado para mí —dijo señalando a una esfera irisada de aproximadamente un metro de diámetro que flotaba en el aire delante de él, unida al resto de sus órganos esclavos por delicadas hilas, a través de las cuales circulaban una especie de nieblas de diferentes colores.

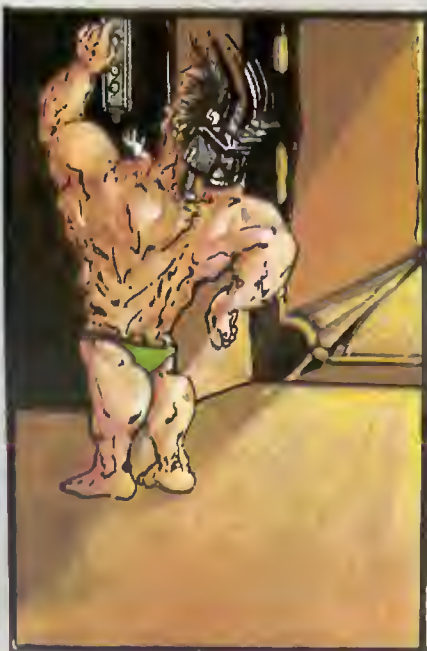
La esfera pulsaba rítmicamente, de forma parecida a un corazón, pero mucho más suave, mucho más... incitante.

A medida que Onerr se acercaba a ella, la esfera pareció tensarse en plena aire, y su superficie se retorcía adaptando diversas formas, cada vez más definidas, mientras una serie de protuberancias comenzaron a alargarse en tentáculos transparentes, que se extendían óvidamente hacia el Danador. El flujo de nieblas de colores desde los órganos esclavos se hizo mucho más intenso.

Onerr observó admirado, como si fuera la primera vez (*bueno, con Madre siempre era la Primera Vez*), o la esfera transformarse definitivamente en una delicada forma ahusada, y observó también, campocida, cómo los tentáculos se desplegaban como la cola de un pavo real y le envolvían tiernamente, adaptándose a la perfección al cantaro hexagonal de su cuerpo.

¡Contacta! Sintió el impacto de la absorbente femineidad de Madre como un puñetazo en la boca del estómago, si hubiera tenido estómago; bueno, tampoco esa impartía demostración.

Se concentró en el contacto con el hula





transparente: Madre era fría, tremendamente fría. Su mente o lo que fuera no albergaba ninguna emoción o sentimiento de ningún tipo. Allí sólo había... una manera de hacer, de actuar; directa y precanebida, según unos reglas.

Las fibras de su propia cuerpo empezaban a vibrar de emoción cuando comenzó a comprender esas reglas, a través de Progie, que podríamos decir que las traducían, desde Madre hasta él. Se sentía cada vez más excitado, cada vez más cerca de la comprensión sintética de algo muy importante, que sólo ella podía darle. Sentía tanta amor por Madre, que ese sentimiento casi ahogaba o la compasión que le producía el hecho de que tal belleza estuviera albergada en el cascarón sin alma de una máquina diseñada por sus antepasadas, como toda lo demás de Tierra. Sentía, ¡Oh Dios, cómo sentía! Entonces llegó Padre.

—ATENCIÓN, PARA SISTEMA FREUD Y SIMBIOPROGRAMA ONERR DE SISTEMA LINCOLN. ESTABLEZCO CONTACTO. PREPARADOS PARA SITUACIÓN LÍMITE. MENEGRAMA DEL SUJETO ONERR EN ALERTA ROJA.

Recibido y procesado, Padre—. Onerr sintió la llegada de Padre como algo infinitamente potente y arrollador, algo que completaba el intercambio de las tres y lo convertía en la Unión, la culminación del Día del Amor. Padre traía con él el Conocimiento.

Le suministró el impulso que a él y a todas las demás le faltaban para comprender el sutil juego de reglas de Madre.

Ahora todo estaba claro; la región Occidental entera de Tierra estaba viva en él y a través de él. Lo comprendía todo: su nacimiento, su cuidadosa educación como

Donador Emocional, la misión de Progie en su médula espinal, era todo eso simultáneamente.

Sus sentimientos eran un torrente que inundaba a Madre por completo, haciéndola brillar con todas las calares del arco iris, iluminándola completamente la Terminal.

En media de la tarmenta, Progie susurró en la mente de Onerr, muy bajito:

—Todavía no lo has comprendido todo, amigo mío. Mira un poco más dentro de Padre. Mira...

Súbitamente arrancada de su éxtasis, sorprendido del «tano» se su simbió, Onerr miró a través de la gloria de luz que era Madre, hacia Padre, y vio...

—ATENCIÓN, Onerr EN ALERTA NEGRA. Situación LÍMITE LO SE LO IMPORTA UNBLEDO.

menegramas en colapso, HA comprendido LA UNIÓN.

¡¡SUJETALO!!

... vía que Madre ya no era fría, sino que sentía el mundo y las cosas con la misma fuerza que él. ¿Con la misma? Con MUCHA más fuerza. Madre ya no era una forma ahusada multicolor, era un ser extraño, bípedo, que se erguía frente a él mirándole con compasión, con infinita pena.

¿Pena? ¿Compasión? ¿Qué significaban esas palabras? ¿Por qué ya no sabía el significado de esas palabras?

Gritó, gritó con toda la fuerza de sus tres brazos, mientras algo dentro de él, descanada y fría, analizaba lo que estaba sucediendo, concluyendo que no había necesidad de gritar; la que ocurría era perfectamente natural.

Onerr dijo:

—IF SITUACIÓN=LÍMITE AND SITUACIÓN=DESCONOCIDA AND

COMPRENSIÓN (MI)=1000 THEN ENLOQUECER (TOTAL)

ENDIF.

Onerr se volvía completamente loca, y se desplomó en el suelo de Terminal como una masa gimiendo desprovista de conciencia, abrazándose sus tres brazos con las flácidas pliegues de su cuerpo hexagonal, ahora ceniciento, de un color gris sucio.

Progie dijo:

—Madre, ¿por qué todas las humanas enloquecen en la Unión? Ya les tengo aprecio, sobre todo a los Donadores Alfa. Son, bueno, son muy cálidos. Sienten.

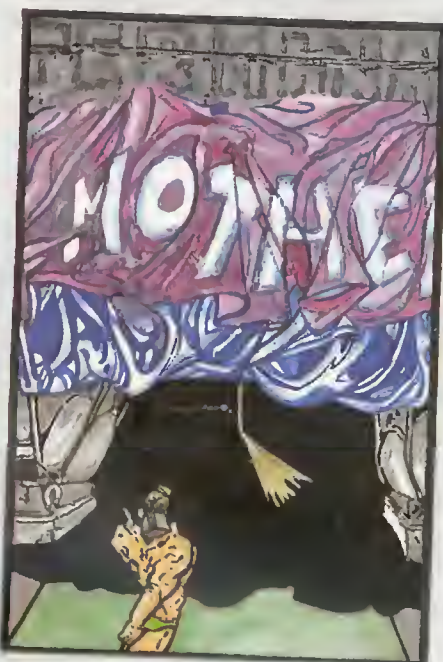
—La serie Onerr no son humanas, simbioprograma, NOSOTROS, Padre y yo, somos humanas. De hecho, somos los últimos humanos, la cumbre de la evolución conjunta Hombre-programa. Al menos, fuimos humanas alguna vez, hace mucho tiempo, cuando podíamos sentir por nosotras mismas, sin la ayuda de la Unión, ni la tuya.

—Ahora, simbio, educarás al siguiente miembro de la serie Onerr y la prepararás para la próxima Unión. Padre y yo lo esperamos impacientes. MUY impacientes—.

Madre pensó, con un risa sarcástica —durante un tiempo y gracias a Onerr podría reír y llorar (¡oh, Dios!, llorar)— en el nombre de sus antepasados daban o los que eran como Ella y como Padre: las llamaban vampiras.

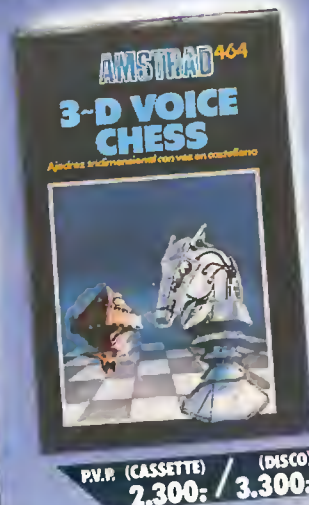
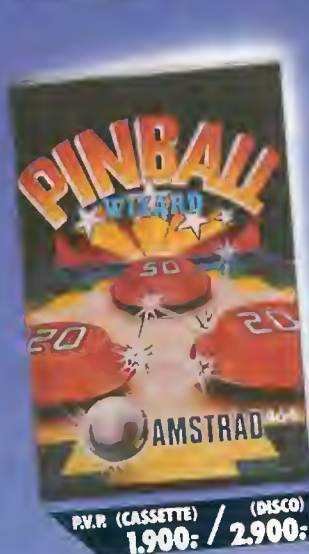
El simbioprograma Onerr se trasladó a la sala de Nacimiento, para implantarse en su próxima huésped. Al hacerla, aliviaría toda lo que hoy había aprendido; para esa la diseñaran.

Bueno, pensó, al menos no era un simbio de los Onerr del Día de la Muerte, porque los humanos tampoco podían morir. Realmente, no impartía demostrado.





# AMSTRAD SOFT



**ACE**

DISTRIBUCION

Actividades Comerciales Electrónicas, S.A.  
Tarragona, 112 Tel 325 15 12. Telex 93133 ACEE E 08015 Barcelona

YA DISPONIBLE EN



... Y EN TODAS LAS  
TIENDAS ESPECIALIZADAS



# Cómo trabaja un profesional

*El es Manolo Orcera, programador. Quizás por el momento su nombre no os sugiera demasiado y sus programas no os sean del todo conocidos. Pero prestad mucha atención a todo lo que nos cuenta en esta interesante entrevista, porque sus palabras son las de un gran programador, y pueden servir de ayuda y ejemplo a todos aquéllos que estéis interesados en el mundo de la programación.*

# E

l primer contacto de **Manolo Orcera** con la informática lo tuvo mientras realizaba el servicio militar. Uno de sus compañeros tenía un ZX 81, y cuando vio por primera vez lo que salía de aquel diminuto aparato, casi le da un paro cardíaco. Le pareció lo más grande del mundo, lo más maravilloso, y desde aquel momento no pudo quitarse de dentro el gusanillo de la programación.

Así, poco a poco, fueron surgiendo sus primeros programas en Basic, sus primera rutinas en código máquina, sus primeras ilusiones.

**—¿Cuándo comenzaste a ver por primera vez la programación como algo factible para ti?**

—La verdad es que no habido ningún momento crucial en el que se haya producido un cambio radical en mi manera de pensar. Desde que comencé a hacer pequeños programas todo se fue sucediendo de una manera muy lineal.

**—Entonces, ¿no estás de acuerdo con la idea tan extendida en la actualidad de que la programación es un tema reservado a las grandes compañías que trabajan con gente muy experimentada y con unos grandes medios?**

—En absoluto. Creo que todo aquél que se proponga llevar a cabo un programa puede hacerlo.

Mira, todos los que están leyendo esta entrevista saben castellano; pues considero que el castellano es bastante más difícil que cualquier lenguaje de programación. Por tanto, cualquiera está capacitado para programar, incluso con más facilidad.

---

**«La primera vez que vi lo que salía de un ordenador, casi me da un paro cardíaco»**

---

**—¿No crees, que sea necesario tener una mente lógica que sea capaz de estructurar y analizar al máximo una idea?**

—No, no lo creo. Lo puede hacer cualquiera mientras tenga algo que comunicar. Ahora bien una cosa es hacer una rutina o un programa y otra bien distinta es «crear» algo.

**—¿Quieres decir con esto que la programación es un arte?**

—La programación es una técnica, y como tal está al alcance de todos. Pero el hacer que eso se convierta en algo agradable, atractivo, original, sí es un arte.

**—¿Por qué te decidiste a dejar el Basic y empezar a programar en Ensamblador, que tiene fama de ser un lenguaje bastante arduo y difícil?**

—Desde luego que es el lenguaje más árido que existe, puesto que no te da nada en absoluto, todo te lo tienes que crear por ti mismo. Pero las ventajas que ofrece son tan abismales con respecto a los demás, que no tienes más remedio

que utilizarlo, y una vez que empiezas, llega un momento en el que ya no puedes prescindir de él.

**—Sin embargo sabemos que en tus programas normalmente utilizas una mezcla de ambos lenguajes. ¿Cuáles son los motivos que hacen que esto sea así?**

—El tiempo. A pesar de que yo personalmente me siento mucho más cómodo trabajando en código máquina, existen muchas ocasiones en las que tienes que recurrir a otros lenguajes de alto nivel, con el fin de ahorrar tiempo, ya que éste es un factor decisivo en la programación.

**—¿Coincides entonces con la tendencia actual de que a pesar de que el ensamblador es una herramienta necesaria, debe utilizarse lo menos posible?**

—Evidentemente todo lo que esté relacionado con el INPUT/OUTPUT de un programa como las rutinas de música, o los scroles de pantalla es necesario hacerlo en código máquina puesto que, por ejemplo, volcar 7K de memoria en 4 centésimas de segundo en pantalla, hasta ahora el único lenguaje capaz de hacerlo es el ensamblador.

Sin embargo, todo lo referente a tratamientos de variables, o de texto, o tratamientos aleatorios, lo

---

**«Lo más importante de un juego son los gráficos. Por desgracia, la programación del mismo no se valora en absoluto»**

---





cantidad de memoria deseada, que normalmente suele coincidir con la tercera o la cuarta parte de los gráficos iniciales.

Las posibilidades del Spectrum,

## EN VIVO

ordenador con el habitualmente trabajamos, son limitadísimas y no nos queda otra opción.

—Bueno, y una vez que tenéis los gráficos, los pocos que os han quedado, ¿qué es lo que hacéis?

—Pensar en la estructura que va a mover esos gráficos. Esta quizás sea la etapa más intuitiva de la programación, pues normalmente te pasas días y días intentando sacar algo positivo y te resulta imposible. Hasta que un día, te vas a la cama, y de pronto, ¡la rutina, la encontré! Te levantas, coges un papel y te pones a escribir. Entonces, a las cuatro o las cinco de la mañana te sientas frente al ordenador, la pruebas y te das cuenta de que no sirve. Te vuelves a acostar y esperas al día siguiente.

—Por lo que dices, parece que usáis la técnica de escribir primero el programa sobre un papel y llevarla después al ordenador. ¿Crees que es la manera más correcta de programar?

puedes hacer perfectamente en un lenguaje de alto nivel, donde no requieres demasiado tiempo y no influyen en el resultado final del programa.

—Estamos seguros que a nuestros lectores les interesaría conocer cuál es el proceso para la creación de un programa. Nos podrías contar cómo se desarrolla esto en un programa cualquiera de los tuyos.

—En general los pasos que seguimos, puesto que trabajo con otro compañero, son muy similares con todos los programas, pero quizás el más espectacular sea el caso del último que hemos realizado: David el gnomo, debido que nos vimos obligados a realizarlos tan sólo en 15 días.

A parte de esto, por regla general, lo que primero solemos hacer es pensar detenidamente en

la idea de cómo queremos que sea el juego. Después estudiamos los gráficos que va a tener y hacemos un cálculo aproximado de la cantidad de memoria que nos van a ocupar. Normalmente desechamos la idea inicial, porque comprobamos que nos ocupa demasiada. Entonces quitamos parte de los gráficos y volvemos a mirar.

Así vamos repitiendo la operación hasta que alcanzamos la

**«El ensamblador es una herramienta necesaria, pero de la que a veces es conveniente prescindir»**

**«Lo que realmente me impulsó a programar fue el que ningún programa me satisfacía totalmente»**

—Efectivamente. Para realizar un programa hay que estructurarlo previamente, realizando por separado cada una de las rutinas en código máquina que se van a encargar de realizar una función específica. Estas se prueban independientemente y posteriormente se estudian en conjunto.

—¿Cuáles son las mayores dificultades con las que os encontráis en la realización de un programa?



—Todos se dividen en tres partes fundamentales: gráficos, programación y sonido. En el caso de David el gnomo, lo que más nos costó fue realizar la estructura de giro de las figuras, pues nos fue bastante difícil dar con algo que realmente nos convenciera a los dos. Pero quizás la pega más grande fue el planteamiento general de la pantalla, pues no conseguíamos ponernos de acuerdo y al final tuvimos que optar por una solución intermedia debido a los grandes problemas de tiempo a los que teníamos que enfrentarnos.

## «Crear un programa atractivo, agradable y original es un arte»

—¿Cuál es para ti la parte más difícil de un juego?

—Los gráficos.

—¿Y la más importante?

—Los gráficos. Esto es sin duda alguna, lo que más llama la atención de la gente. La parte de la programación de un juego puede estar bien o mal realizada, eso por desgracia, no le preocupa a nadie. El caso es que el juego funcione y que resulte vistoso.

—Ahora nos gustaría hacerte un rápido cuestionario, al que te rogamos contestes lo más brevemente posible. Ordenador favorito.

—Amstrad. Es el que ofrece una



mayor relación calida/precio en todos sus modelos, excepto en el 256 que me parece una auténtica castaña.

—Ordenador con el que te gustaría trabajar.

—Hasta ahora lo he venido haciendo con un Spectrum, pero pienso cambiar a Amstrad, no sólo por sus características sino porque a sus usuarios les tengo un cariño especial; respetando a la gente de Spectrum, donde sé que hay verdaderos viciosos de la informática, la de Amstrad me parece mucho más seria, que saben mejor lo que quieren.

—Programa favorito.

—En arcade, en cuanto a gráficos y presentación Highway Encounter, pero en cuanto a genialidad programativa, sin duda alguna Alien 8. En lo relativo a juegos para mentes flácidas y no de dedillo ágil, me quedo con mi Pentac, ya que cuando lo hicimos pretendimos que fuera un auténtico test de inteligencia y creo que lo hemos conseguido.

## «El software pirata lo único que va a conseguir es que nadie haga más programas»

—¿Qué te sugiere la pantalla Ultimate?

—Son las mejores. Con Alien 8 sacudieron totalmente la técnica de la programación. Me parecen maravillosos.

—Probabilidades de vivir de esto.

—Absolutamente ninguna. Los programadores somos programadores y no comerciantes. Somos carne de software. Puedes dedicarte a esto porque te encante, pero desde luego no recomiendo a nadie ser programador para ganarse la vida.

—Y, ¿qué nos puedes contar en cuanto a futuros proyectos?

—Pues tenemos en mente varios juegos. Dejando a un lado los puzzles, que es casi lo único que hemos hecho hasta ahora, vamos a pasar a hacer arcades, uno de guerra en el espacio y otro de guerra terrestre, con luchadores salvajes y espadas y mazas y cosas de esas, que según creo va a gustar mucho. Este es el juego al que siempre me hubiera gustado jugar y que nadie hizo.

Ahora que hemos tocado este tema, creo que esta es la verdadera razón que me movió a realizar mis propios programas, el que mis apetencias no estuvieran satisfechas; porque aunque hay un software excelente, aún no he conseguido encontrar un juego que verdaderamente me emocione.

—Y ya para finalizar, nos gustaría que dieras un pequeño consejo a todos los jóvenes que están comenzando a programar y que tienen puestas gran parte de sus ilusiones en esto de la informática.

—Pues quisiera decir, aunque vaya en contra de mi economía y proyectos de futuro, que cojan las cintas de juegos que tienen y las guarden en un cajón, y que sepan que debajo de sus deditos tienen la mejor calculadora científica que Einstein jamás pudo soñar y que ellos la tienen prácticamente abandonada.

También creo que es muy importante que piensen que cuando compran un programa, están adquiriendo el trabajo y el sudor de muchas personas y que es bastante miserable el que cierta gente te ofrezca por veinte duros ese programa que tanto ha costado realizar. En una palabra, que dejen de comprar software pirata, que lo único que hace es destruir y con eso van a conseguir que nadie pueda seguir programando.





# STARmouse

para los siguientes ordenadores

amstrad  
commodore  
QL y próximamente  
expectrum



## STARmouse

### PURICORP

C/ CALNUEVAS, 4  
19003 - GUADALAJARA  
ESPAÑA

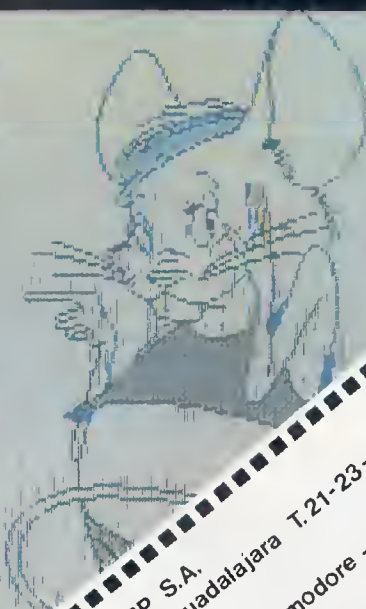
DISEÑO DISEÑADO POR:  
ALFONSO RUIZ Y CARLOS BARRA

PROGRAMA HECHO POR:  
ABEL RUIZ LOPEZ

DE JUEGOS REALIZADOS POR:

TIAGO BARREIRA  
ABEL RUIZ "ACL"

JUAN ANTONIO RODRIGUEZ "JOHAN"



Haz tu pedido a PURICORP S.A.  
c/ Calnuevas 4 19001 - Guadalajara T.21-23-58 (911)

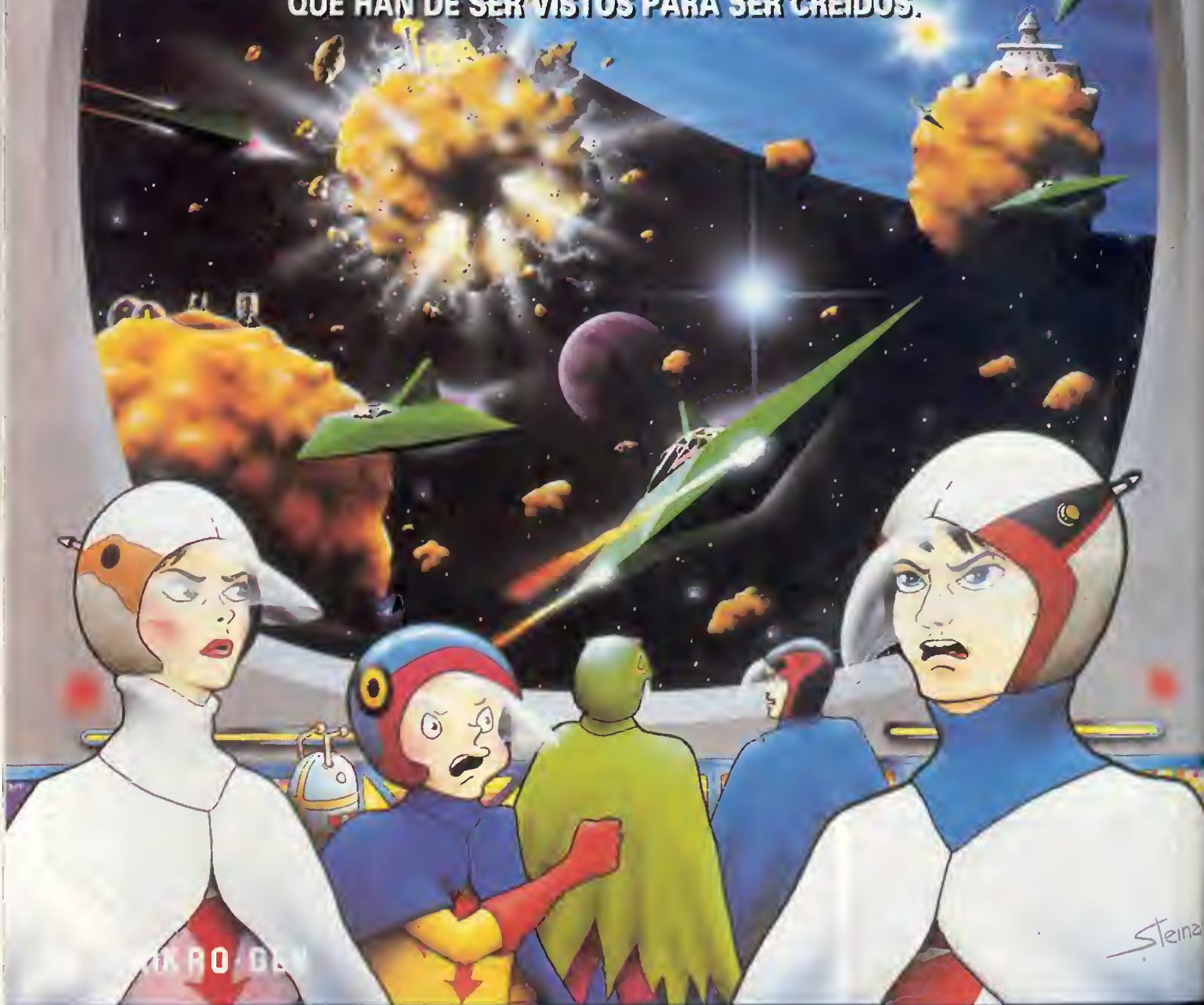
Precios Amstrad o Commodore - Disco  
13.800 - pts. IVA. incluido  
Amstrad o Commodore Cinta  
11.500 - pts. IVA. incluido



SI BUSCAS LO MEJOR **ERBE** Software LO TIENE

# BATTLE OF THE PLANETS

UNETE AL "COMANDO G" EN SU ULTIMA AVENTURA CONTRA ZOLTAR  
EN UN MARAVILLOSO JUEGO REALIZADO CON UNOS GRAFICOS TRIDIMENSIONALES  
QUE HAN DE SERVISTOS PARA SER CREIDOS.



**BATTLE OF PLANETS ES EL PROGRAMA SELECCIONADO PARA EL  
CAMPEONATO INTERNACIONAL DE JUEGOS DE ORDENADOR.  
CONSIGUE LA MAXIMA PUNTUACION ESPAÑOLA Y PARTICIPA EN LA FINAL DE LONDRES.  
(LAS BASES DEL CONCURSO ESTAN EXPLICADAS EN EL PROGRAMA)**

DISTIBUIDOR EXCLUSIVO PARA ESPAÑA ERBE SOFTWARE C/. STA. ENGRACIA, 17. 28010 MADRID, TFNO.: (91) 447 34 10  
DELEGACION BARCELONA, AVDA. MISTRAL, N.º 10 - TFNO.: (93) 432 07 31



# DURELL

## SABOTEUR!



SPECTRUM 48K  
R.R.P.  
£8.95

COMO EXPERIMENTADO MERCENARIO CUIDADOSAMENTE ENTRENADO EN ARTES MARCIALES, DEBES CUMPLIR LA MISION QUE TE HA SIDO ENCOMENDADA: ROBAR EL DISCO QUE CON LA LISTA DE LOS REBELDES TIENE EL GRAN DICTADOR.

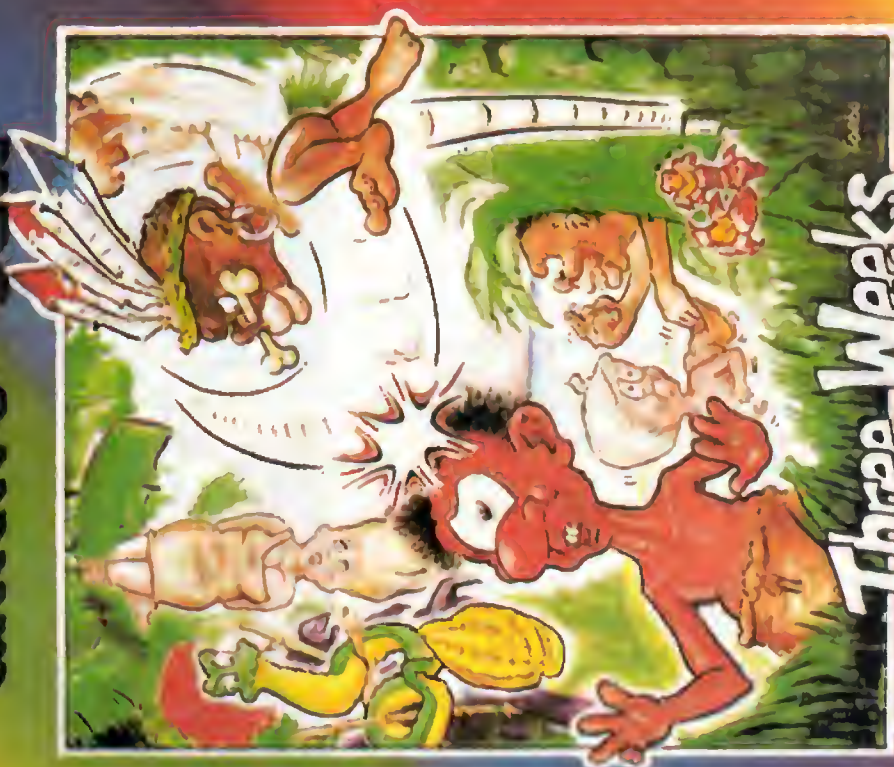
# SI BUSCAS LO MEJOR



Software

# LO TIENE

SENSATIONAL SOFTWARE FROM  
**MIKRO-GEN**



## Three-Weeks in Paradise

LA FAMILIA WALLY ATACA DE NUEVO EN LA MEJOR Y MAS DIVERTIDA AVENTURA QUE PUEDAS IMAGINARTE. TODOS LOS PELIGROS Y COLORIDO DE LA SELVA EN TU ORDENADOR.



# LA INFORMATICA EN CASA

**No hace mucho tiempo que los ordenadores eran, para la mayor parte de nosotros, grandes máquinas fantásticas y desconocidas. Se les conocía por haberlos visto en el cine o porque alguien nos había contado las maravillas que eran capaces de realizar.**

# S

obíamos que eran capaces de realizar cálculos complicadísimos muy rápidamente y agilizar y simplificar ciertos tipos de trabajos que manejaban una gran cantidad de datos. Y en cualquier película o novela de ciencia ficción que se preciaran no faltaba un robot inteligente que siempre ayudaba a los buenos.

Pero pocos datos más conocíamos. Y, sobre todo, pocos datos reales.

Todos estábamos de acuerdo que eran unos inventos maravillosos pero, **¿quién se atrevía a meter un ordenador en casa?** O mejor, **¿para qué necesitábamos semejantes armatostes dentro de nuestro hogar?**

## La evolución de los ordenadores

Y de pronto, casi como por arte de magia, los ordenadores encogen, se hacen mucho más pequeños y manejables, se meten en nuestros vidios y se convierten en algo poco menos que imprescindible. Hemos perdido el miedo que nos recorría el cuerpo con sólo nombrarlos y comenzamos a considerarlos como uno herramienta más de los que tenemos en casa.

Pero nos surge una pregunta: **«¿qué hace una máquina como tú en unas casas como las nuestras?»** (adaptación in-

formática de una conocida frase cinematográfica). Es decir, ¿para qué podemos utilizar un ordenador en nuestro dulce hogar?

Ante todo vamos a ser sinceros: nos hemos comprado un ordenador poro **jugar**. Esto es la primera utilidad que casi todos le hemos encontrado, aunque a lo largo de lo comprado nos autopersuadiéramos de que la necesidad básica era otro. Lo que más nos llamaba la atención era el poder transportarnos a un mundo diferente donde lo mismo podíamos estar conduciendo a gran velocidad un bólido de carreras o pilotando uno nove **hiperespacial**.

Al principio los programas de juegos para los ordenadores **caseros** estaban basados, o mejor trasvados, de los máquinas recreativas de los **bares** o de las salas de juegos. Todo lo que disponíamos para nuestro uso doméstico se reducía a los clásicos **marcianitos**, los «**comecocos**», los fotitos choques de **asteroides** en tres dimensiones y hasta podíamos echar un partido al tenis estando representados en la pantalla por un simple rectángulo que utilizábamos o modo de roqueta.

Recordemos que no se trataba de juegos muy complicados. En ellos primaba el atractivo del color y el sonido y su finalidad no era otro, sino poner a prueba la rapidez de reflejos del usuario.

## Cada vez, más y más ordenadores

La proliferación del ordenador personal tuvo como consecuencia que se abandonara en parte el uso de las máquinas de los bares, puesto que en casa podíamos disponer de los mismos juegos y usarlos con más comodidad, sin mirones y todo el tiempo que quisiéramos. Por tanto los programadores tuvieron que in-

M. Barco







geniar juegos cada vez más sofisticados, aprovechándose de la mayor capacidad de memoria de las máquinas de uso público, para atraer a los usuarios, hasta llegar a la belleza de los programas que se ven actualmente y que parecen reales, como si se trataran de películas.

Por otro lado también el catálogo de juegos para los ordenadores caseros debió ampliarse y diversificarse: una vez que se dirige una nave o un **comecocos** es como si ya se hubieran dirigido todos.

Los usuarios cada vez exigían mayor perfección y variedad. Por lo tanto, no sólo se trataba de encontrar nuevos temas, sino que éstos debían ser lo más originales posible. Todo ello unido a una gran perfección de movimientos, la espectacularidad del sonido y una puesta en escena, cada vez más brillante. Las pantallas se multiplican, se diferencian entre sí, los programas se complican, se hacen cada vez más atractivos y absorbentes.

### La batalla de la originalidad

Y algo nuevo. Ya no se busca solamente la rapidez de reflejos, sino que se intenta atraer a un mayor número de consumidores a quienes interesa más la reflexión y el desarrollo del propio poder de decisión. De aquí nace la necesidad de los juegos de **aventuras**. Incluso se hacen adaptaciones de otros medios: cine, cómic, televisión, literatura. Se buscan héroes propios que pasen por cien mil pruebas. Viajamos con ellos a través de cualquier tiempo descubriendo que no sólo el futuro nos reserva cualquier tipo de heroicidades.

Paralelamente a estas **aventuras** se siguen desarrollando los juegos de **mesa** clásicos que también buscan la intervención del jugador. Generalmente admiten la posibilidad de jugar con la máquina. ¡Es un gran reto enfrentarse con la técnica y lograr vencerla!

Podemos llegar o pasar una buena y relajada tarde de una forma clásica —*son juegos de toda la vida*— pero esto vez con la gran ayuda de un ordenador.

También podemos desarrollar nuestra lógica poniendo en marcha un juego de **estrategia**. En ellos se simula una acción real —*una batalla, por ejemplo*— en la que nosotros podemos intervenir variando el desarrollo de la misma tomando en cada momento la decisión que creemos más acertada según la situación del juego. **¿Recuerda la película Juegos de guerra?**

Hasta aquí diremos que hemos conseguido tres logros con el hecho de tener un ordenador en casa:

- Nuestros reflejos pueden estar a punto.
- Nuestra mente analítica ha encontrado un método muy divertido para ejercitarse.
- Gracias a nuestras mayores exigencias

estamos obligando a los programadores (*quizá a nosotros mismos*) a realizar verdaderos alardes de imaginación e ingenio y a estar siempre al día.

Y todo esto tiene una consecuencia también importante, aunque ya no personal.

Se está creando una gran industria informática, cada vez más amplia y fuerte, sustentada sobre este mundo de diversión y entretenimiento. ¡Quién sabe si no es la solución al futuro individual de alguno de ustedes!

### ¿Sólo para jugar?...

Sin embargo, el juego no es el objetivo primordial de un ordenador. En tan sólo un pelotazo, quizá el primero y el más divertido, de los muchos que haya que subir para adentrarnos en el complejo mundo de la informática. No olvidemos que nuestro pequeño amigo es una nave que nos unirá con el futuro, luego estamos en la obligación de conocerle mejor y de exigirle que nos ofrezca algo más que un rato de diversión.

Alguien dijo que lo realmente bueno del ordenador es que se trata de una herramienta didáctica y divertida. Podemos estar aprendiendo continuamente cosas sobre él, ya que generalmente excitará nuestra curiosidad y entraremos en un círculo vicioso que nos obligará a seguir investigando todas y cada una de sus posibilidades.

Vamos a hacer hincapié en este punto. El ordenador vive en el futuro y de ahí la importancia de que todos, pero sobre todo la infancia, aprendamos a convivir con él, a manejarlo con soltura, a verlo como una máquina capaz de ayudarnos en nuestra tarea de formación personal.

Se ha dicho que el ordenador acabará por sustituir a los libros. Quizá sea una frase demasiado fuerte y rotunda, pero lo que sí está claro es que puede llegar a ser el ayudante ideal para cualquier tipo de enseñanza.

Esto es algo que ya han comenzado a ver algunas de las más importantes editoriales que se dedican a publicar tradicionalmente libros de texto, y que ahora están sacando al mercado una serie de programas didácticos muy buenos que cumplen perfectamente la misión de apoyar y extender las explicaciones de los profesores, a la vez que los estudiantes pueden verlo todo de un modo más gráfico y pueden experimentar con ello.

De esta manera, el estudio no consiste ya en aprender una serie de datos fríos y distantes, sino en ver y manejar cada uno de los conceptos que nos van explicando. Ha pasado a ser una actividad **viva**.

Es lo que conocemos como «**enseñanza asistida por ordenador**». Así podemos conseguir que las «horas a codos» se conviertan en momentos agradables, entretenidos y, sobre todo, mucho más creativos.



## ...no, también sirven para aprender

Como prueba, ahí va un dato: para las generaciones anteriores las **matracas** era un **ladrillo** —salvo honrosísimas excepciones— mientras que los estudiantes de hoy en día ven en las Matemáticas que nos cuenta el ordenador la asignatura más apasionante de cuantas existen.

Por otro lado, cualquier padre que se precie habrá intentado, y en la mayoría de los casos conseguido, hacer un programa sencillito que enseñe a sus hijos inglés, la tabla de multiplicar o la ortografía de cualquier palabra castellana. También en este caso el padre está aprendiendo algo.

Lo mismo que los juegos, los programas didácticos son un reclamo a la hora de utilizar un ordenador. Mientras jugamos o estudiamos Matemáticas mediante un buen programa sin darnos cuenta nos vamos adentrando poco a poco en este novedoso mundo y comenzamos a tomarle gusto a este **aparato** recién llegado a casa.

Y, generalmente, así ocurre. Si queremos utilizar un ordenador para realizar cualquier actividad se supone que necesitamos un conocimiento, aunque sea mínimo, de su manejo. Paulatinamente estas nociones básicas van convirtiéndose en un deseo de profundización en esta nueva herramienta que hemos conseguido. Y aquí el campo recién abierto es vastísimo, tanto que casi no nos atrevemos a enumerar las ramificaciones que puede llegar a tener.

Podemos dividirlos en dos grandes grupos:

- Los apasionados de la electrónica encuentran un camino sin fin en el que cuanto más se avance más apasionante se vuelve lo que nos queda por recorrer. Y no dudamos que el futuro de este terreno se prevé atractivo al máximo.

- Quienes prefieren comunicarse con el ordenador tienen en sus manos una labor de investigación única: lenguajes, trucos, programación y cualquier otro campo que pueda interesarle. El ordenador ha abierto nuevos horizontes a nuestro interés por aprender por amplio y disperso que pueda parecer.

Dentro del primer grupo se encuentran los que dejando volar su imaginación piensan, por ejemplo, en la cantidad y cantidad de nuevas técnicas que pueden encontrar para que su ordenador casero sea capaz de manejar con precisión matemática cualquier aparato eléctrico. ¿Qué tal si intentamos controlar que todos los semáforos de nuestra maqueta de tren funcionen de forma que no se produzcan choques ni descarrilamientos irreparables? ¿Por qué no intentar fabricar un controlador que nos permita aplicar a nuestro cámara fotográfico o a nuestro pequeño tomavistas las técnicas utilizadas en las grandes superproducciones de Hollywood?

Ya existen en el comercio pequeños contro-

ladores domésticos que podemos aplicar a nuestro nuevo amigo. Con ellos podríamos encender y apagar las luces de nuestra casa a unas horas determinadas, poner en funcionamiento la calefacción cuando hace frío o fabricarnos un sistema de seguridad digno de cualquier historia policiaca y muchísimas cosas más que aunque de momento puedan parecernos sencillas, con paciencia y buena voluntad podremos ir perfeccionando e incluso ampliando.

Por otra parte, los amantes de las relaciones sociales, a cualquier nivel, podrán aprender a comunicarse con un ordenador «**case-ro**» en cualquiera de los lenguajes empleados hasta ahora solamente en los grandes ordenadores o sistemas informáticos.

De hecho ya existen versiones de compiladores y traductores realizados para varios de estos lenguajes de alto nivel que podemos utilizar de inmediato en nuestro ordenador. Pero lo realmente bueno es que las casas de software no descansan y están dispuestas a proporcionar los útiles necesarios para que podamos programar hasta en los lenguajes más avanzados.

## Creatividad, ordenadores y arte

Y aquí hemos llegado a otro de los motivos por los que el ordenador ha entrado en casa: la necesidad de **aprender a programar**. No tenemos por qué esperar a que otros nos den la **aventura** ya hecha. Vamos a inventarla por nosotros mismos.

Con un mínimo interés y a pesar de bastantes ensayos fallidos seremos capaces de mejorar cualquier programa que podamos encontrar en un libro. O mejor todavía, llegaremos a conseguir que el juego de nuestros sueños se convierta en realidad siendo nosotros sus diseñadores y autores. No lo duden, es preferible dar este paso decisivo y no conformarnos con utilizar los programas de los demás.

Hasta para algo tan aporatamente contradictorio como es el Arte puede sernos muy útil nuestro pequeño amigo. Vamos a conseguir que el ordenador y el Arte se den la mano —no tienen por qué ser incompatibles.

Existen muy buenos programas que nos van a permitir empezar por unos sencillos trazos que pueden luego rellenarse de colores y complicarse, hasta convertirse en un juego de masas y formas.

Así es como nace la pintura: experimentando con colores, masas y formas. No es necesario ser un dibujante genial ni tener un inspiración extraordinaria para ser capaces de realizar una buena composición.

Y cuando por medio del **juego** hayamos aprendido a dibujar tendremos en casa la posibilidad de dedicarnos o algo que es a la vez artístico y técnico: el diseño. **¿Quién sabe**



## hasta dónde podemos llegar en este campo?

Con unas nociones de dibujo seremos capaces de crear espacios tridimensionales y, una vez logrado, será más fácil llenarlos con nuestras fantasías: maquetas, formas... **¿Qué tal si cambiamos la decoración de nuestra casa sin mover ni un solo mueble?**

Y si después resulta que somos unos **manitas**, ¡manos a la obra!, porque con la ayuda del ordenador tenemos a mano todos los dibujos, las medidas, los pesos. En fin, todos los detalles técnicos que nos sean necesarios. Nuestra maqueta de avión volará, nuestro barco no se hundirá y algún que otro armario encajará perfectamente en ese hueco libre y no en aquél otro como creíamos en principio.

Incluso podremos ver nuestro jardín en una pantalla antes de realizarlo para saber si las petunias tendrá poca luz o si la mancha blanca de las margaritas destocaría más delante del verde oscuro de las aspidistras.

El que las grandes fábricas usen programas para diseñar sus productos no debe dejarnos impasibles o desanimarnos. Muy al contrario, debe servirnos de estímulo porque nosotros también podemos contar con la colaboración de una «**máquina**» que nos va a permitir probar o modernizar cualquier cosa, quizá la carrocería de ese coche que usamos para cargarlo de trostos y meterlo campo a través o





## Las aplicaciones «serias» también tienen derecho a la vida

Quién más, quién menos todos tenemos a nuestros familiares y amigos localizados en un detallado agenda informática. Es quizá la primera «**aplicación**» que se nos ha ocurrido. El ordenador nos puede ayudar a suprimir larguísima ficheros que deben ser puestos al día con añadidos y correcciones. Una ficha en cinta o en disco siempre está limpia y sin tochos como el primer día, por muchos correcciones que hoyamos hecho.

Con alguno oportuno modificación o este programa podemos encontrar un magnífica fichera que nos ayude a ordenar nuestra biblioteca y localizar esos libros prestados que ya damos por perdidos. Y quien dice biblioteca puede decir discoteca, videoteca o nuestras queridas y viejas colecciones de sellos, cálices o cromos.

Y de esto o tener una clasificación de clientes y acreedores de nuestro posible negocio familiar sólo hoy un pequeño paso.

Con un programa de Base de Datos, nuestro ordenador podrá implantar en cosa un orden y una eficacia propio de los más moderno oficina. Y podemos permitirnos el lujo de ser despistados sin quedar mal ya que puede recordarnos el mínima data, cualquiera que sea: una fecha, un nombre, una editorial, una dirección, etc...

Y más todavía. Si o la hora de escribir una carta nos da una pereza insuperable, podemos utilizar un **tratamiento de textos**. Con él, nuestro ordenador nos permitirá escribir cualquier texto sin que queden en el papel marcas de errores que hayamos cometido. Una simple corrección de lo escrito en la memoria y basta. También podremos intercalar o borrar líneas, poner cobeceros de informes, etc., todo ello con la más absoluta limpieza.

Además con un periférico apropiado y con ton sólo indicárselo socorremos cuantas copias queramos de un determinado texto. Podemos decir que hemos encontrado un buen **bolígrafo** para escribir todas esas felicitaciones de Novidad o las cartas comerciales de un determinada negacia a segura que cientos de cosas más.

Vamos a dar un giro en las prestaciones y ocupémonos de un **negro** problema: llegar o final de mes con nuestra pobre sueldo.

Podemos utilizar nuestro ordenador como si se tratara de uno gestor personal. Introduciéndole los datos necesarios puede prepararnos los presupuestos mensuales, llevarnos la contabilidad y hacernos **atterrizar** cuando sañemos con esa cadena de sonido, esa parcelita en la sierra o un viaje a las Bahamas. O tal vez todo lo contrario.

Camo una vez hecho un buen programa lo mismo puede servir para aplicarlo o uno co-

so o un negocio, ya tenemos otro aplicación más. Claro que, caso de no tener negocio tal vez tengamos un amigo a vecina a quien podamos hacer un favor.

Y, en letra pequeña, para que no se entere nuestra gestaría, **¿se ha dado cuenta que el ordenador es una calculadora eficaz y potente?** A la hora de realizar la declaración que ya sobemos puede superar quien va a sernos de una utilidad inimaginable.

Pero no sólo es en ese desgraciado momento. Un calculadora con las posibilidades de nuestro ordenador cosera que utilice una buen **hoja de cálculo** puede ser utilizado para realizar las operaciones necesarias dentro de esas utilidades que en algún momento todos hemos necesitado.

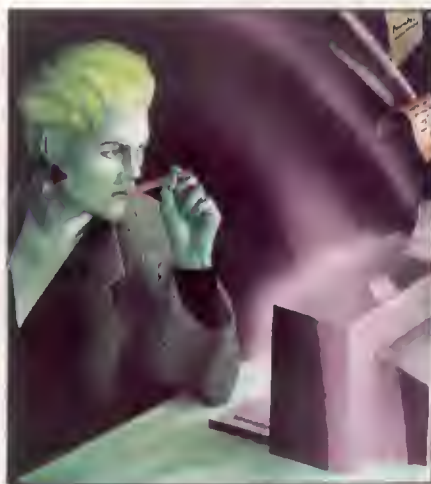
Ahora que, finalmente, hemos comprobado que cuerpo no hay más que una, y que cuando se nos termine se acabá, es necesaria tener conciencia de la importancia de una alimentación equilibrada y una dieta sana.

Es el momento de **cargar** al ordenador con nuestros problemas y dejarle que administre nuestras proteínas, calorías, hidratos, etc... Que vigile nuestro peso, nuestra necesidad de practicar deportes, nuestras biorritmos...

Lo que comenzó siendo «**una máquina de jugar**» se ha convertido en una gran herramienta que va a sernos de gran utilidad ahora y en el futuro. Con un programa personal (que ya supondría un primer trabajo) podemos estar seguros de que vamos a encontrar solución a muchas de las problemas que ahora nos parecen inaccesibles. **¿Ha oído hablar de la inteligencia artificial?**

Imaginamos que muchas de las sugerencias que nos permitamos hacerle ya habrían pasado por su imaginación, en este caso nos gustaría servirles de aguijón: no espere más, actúe. Si lo que hemos hecho ha sido proporcionarles unas cuantas elementos de juicio para buscar la respuesta a la anteriormente mencionado fuese cinematográfica:

**¿Qué hace una máquina como el ordenador casero en unos hogares como los nuestros?** nos alegramos muchísima y ahora... prepárese para el futuro.



poro aprovechar correctamente los puntos de luz que tenemos en una habitación.

**¿Y por qué no hacer feliz a su esposa?** Que elija ropa de alta costura, tallas, medidas y hechuros favoritos y el ordenador se encargará de diseñarlo. O mejor todavía, que aprenda ello o construir el programa necesario. ¡Dios, tus días están contados!

Y lo mismo puede decirse de lo músico. Ya no es necesario tener un piano de cola en caso poro que podamos dedicarnos o cualquier labor musical, ya sea interpretativo o incluso de composición.

Estamos de acuerdo en que no es lo mismo —y no lo será nunca— pero de lo que se trata es que cualquiera que se sienta oído por la música encontrará en su ordenador personal un perfecto colaborador dispuesto o oceptor cuantas pruebas, ensayos, experimentaciones queramos realizar con él, por extraños y poco académicas que resulten.

El ordenador se ocupará, mediante un programa de los ya existentes o alguno nuevo que usted diseñe, de la parte mecánica y tedioso y más ógil. Y como en el caso del dibujo, no se precisa ser un músico genio poro conseguir algunos tonalidades agradables al oído.

Pero por si hay alguien que piense que el ordenador ha de usarse para algo más práctico na le defraudaremos. Todos los problemas de gestión pueden ser resueltos mediante nuestro amigo el «**pequeñajo**».



# MICRO-1

Duque de Sesto, 50. 28009 MADRID. Tel. (91) 275 96 16/274 53 80  
(Metro O'Donnell o Goya)

EL IVA  
LO PAGA  
MICRO-1



Disco Cumana 5.25": **47.900** ptas.



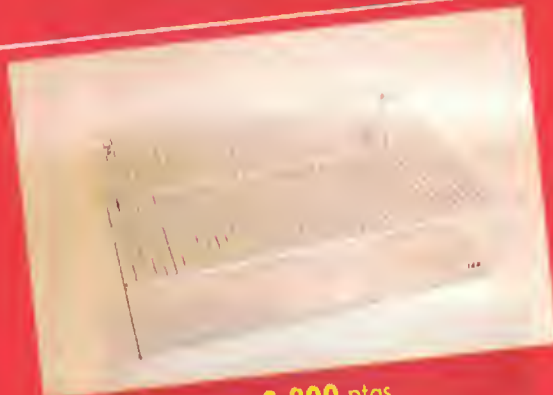
Quick Shot II: **2.395** ptas.



Quick Shot V: **2.995** ptas.



Lápiz óptico: **4.850** ptas.



Soporte impresora: **3.800** ptas.



Amplificador/ Sintetizador de voz: **7.900** ptas.

Diskette 3": **995** ptas.  
Cinta C-15 especial: **85** ptas.

Precios excepcionales para tu AMSTRAD  
464: **6.128** ptas. y **8.256** ptas.

Impresoras: **20%** dto.  
sobre **P.V.P.**

Pedidos contra-reembolso sin ningún gasto de envío. Teléfonos  
(91) 275 96 16/274 53 80 o escribiendo a MICRO-1. Duque de Sesto, 50. 28009  
MADRID



# CONTROL DEL IVA

**La entrada en vigor del Impuesto sobre el Valor Añadido en enero de este año, ha traído consigo la obligatoriedad de llevar un control exacto de las facturas de proveedores y de clientes, a los que respectivamente hayamos pagado o cargado dicho impuesto dando un número de orden a dichas facturas. Eso es lo que hace este programa.**

El programa en grandes rasgos, es como sigue:

En las líneas 60-80 vienen los datos para crear el fichero con el programa RANDOM-F.BAS.

Las líneas 160-400 establecen las pantallas de texto, y presentan el menú general.

A partir de la línea 410 empiezan las opciones 1 y 3 del menú. Primero inicializa el fichero si no lo está todavía, y lee el primer registro para saber el siguiente número de apunte. Luego presenta la pantalla principal, en la que pregunta primero la opción. Permite cuatro respuestas:

Altas, Bajas, Modificaciones y Consultas. En las altas, da el número de apunte, y si no le da, pregunta el N.º de factura, y busca dicho número secuencialmente. La fecha, hay que introducirla dando día, mes, año y pulsando luego ENTER. Si el IVA hubiera que darlo con decimales, no hay que poner coma sino PUNTO decimal.

En las líneas 960-1110 graba el registro en el disco, y actualiza el primer registro con el próximo número de apunte.

Las líneas 1510 a 1600 dan de baja un registro. No borran el registro, sino que ponen el campo Estado con una «B». Esto hace que luego no sean tenidos en cuenta a la hora de Listar o hacer liquidación.

**E**stá totalmente realizado en Basic, y utiliza un fichero de tipo directo, lo que hace necesario tener en el mismo disco además de este programa, una copia del programa RANDOM-BIN que viene en el disco de regalo.

El primer registro del fichero es utilizado como índice, almacenando el número del último apunte realizado. La primera vez que se utilice el programa inicializará el fichero colocando en dicho registro el primer momento de apunte. El diseño de registro es el siguiente para los registros desde el 2 en adelante:

NOMBRE	POS. INICIAL	LONGITUD	DATOS
Estado	1	1	A. Alta B. Baja
N.º Apunte	2	3	Númericos
N.º Factura	5	7	Alfanuméricos
Importe	12	7	Númericos
Porcentaje IVA	19	4	Númericos (2 dec.)
Fecha	23	6	Númericos
Tipo	29	1	S. IVA Soportado R. IVA Repercutido

*Serie*  
**ORO**



A partir de la línea 1610 empiezan las opciones 2 y 4 del menú. Proporcionan un listado por orden de apunte de las entradas habidas entre la primera y la última fecha dadas. Se puede obtener el listado sólo por batalla, o también por impresora.

En la línea 2110 empieza el proceso de acumulación. Pide la primera y la última fecha. Por pantalla nos dará sólo el total de retenciones soportadas y repercutidas, pero si pedimos resumen por impresora, listará todos los apuntes que estén entre dichas fechas, y las sumas totales.

Cuando se esté tecleando el programa, hay que tener en cuenta que las líneas que contengan CHR\$ son códigos de control de la impresora. Aunque suelen ser los mismos en todas las marcas, algunos pueden diferir y habrá que cambiarlos o suprimirlos.



```

10 '*****
20 ' I.V.A.
30 ' @ MICROHOBBY AMSTRAD
40 ' F.J.B.T.
50 '*****
60 'No.fichas= 999
70 'Long.Reg = 29
80 'Nombre reg.:IVADATOS
90 '-----
100 ' CARGA RANDOM.BIN
110 '-----
120 IF PEEK(&9C00)=1 THEN 160
130 MEMORY &9BFF
140 LOAD "random",&9C00
150 CALL &9C00
160 '-----
170 ' MENU GENERAL
180 '-----
190 CLS:MODE 2
200 WINDOW #1,1,80,1,3
210 WINDOW #2,1,80,4,4
220 WINDOW #3,1,80,5,22
230 WINDOW #4,1,80,23,23
240 WINDOW #5,1,80,24,25
250 PRINT #2,STRING$(80,"=");:
260 PRINT #4,STRING$(80,"=");:
270 LOCATE #1,27,1:PRINT #1,"C O N
T R O L D E L I.V.A."
280 LOCATE #1,26,2:PRINT #1,"=====
=====
290 LOCATE #3,30,2:PRINT #3," M E N
U G E N E R A L "
300 LOCATE #3,30,3:PRINT #3,"=====
=====
310 LOCATE #3,27,5:PRINT #3,"1.- Re
gistro I.V.A. Soportado."
320 LOCATE #3,27,7:PRINT #3,"2.- Li
stado I.V.A. Soportado."
330 LOCATE #3,27,9:PRINT #3,"3.- Re
gistro I.V.A. Repercutido."
340 LOCATE #3,27,11:PRINT #3,"4.- L
istado I.V.A. Repercutido."
350 LOCATE #3,27,13:PRINT #3,"5.- P
roceso de Liquidacion."
360 LOCATE #3,27,15:PRINT #3,"6.- F
in de Programa."
370 LOCATE #5,31,2:INPUT #5,"Numero
de Opcion (1..6) =>";OP
380 IF OP<1 OR OP>6 THEN SOUND 3,20
0:CLS #5:GOTO 370
390 IF OP<=2 THEN OPCIF="Soportado
" ELSE OPCIF="Repercuto."
400 ON OP GOTO 410,1620,410,1620,21
20,2790
410 '-----
420 ' REGISTRO I.V.A. s y r
430 '-----
440 REG$="IVADATOS":a$=""
450 !OPEN,& reg$,1,29,1
460 !READ,& a$,1,1
470 CLS #3:CLS #5:
480 c$=LEFT$(a$,1)
490 nap$=MID$(a$,2,3):nap=VAL (nap$)
:nr=nap+1:ulap=nr
500 FL$=MID$(a$,12,6)
510 IF c$="*" THEN GOTO 550
520 '*** INICIALIZACION DE FICHERO
530 a$="001
"
540 !WRITE,& a$,1,1
550 '*** FIN DE INICIALIZACION
560 CLS #3:CLS #5:CLS #1:
570 IF OP=1 THEN LOCATE #1,21,1:PRI
NT #1,"R E G I S T R O I.V.A. S
O P O R T A D O "
580 IF OP=3 THEN LOCATE #1,19,1:PRI
NT #1,"R E G I S T R O I.V.A. R
E P E R C U T I D O "
590 LOCATE #1,18,2:PRINT #1,"=====
=====
600 '-----
610 ' PANTALLA PRINCIPAL
620 '-----
630 CLS #3:op$=""
640 LOCATE #3,20,4:PRINT #3,"OPERAC
ION (A,M,B,C)...: "

```

```

650 LOCATE #3,20,6:PRINT #3,"AFUNTE
Numero ...: ..."
660 LOCATE #3,20,8:PRINT #3,"FECHA
FACTURA D,M,A...:"
670 LOCATE #3,20,10:PRINT #3,"No. F
ACTURA (7X)...: ..."
680 LOCATE #3,20,12:PRINT #3,"IMPOR
TE (7N)...: ..."
690 LOCATE #3,20,14:PRINT #3,"I.V.A
(NN,N)...: ...%"
700 LOCATE #3,20,16:PRINT #3,"TOTAL
":opc1$:" ...: ..."
710 LOCATE #3,45,4:INPUT #3,"",op$:
op$=UPPER$(op$)
720 IF op$="" THEN op$="N"
730 IF INSTR("ABMC",op$)=0 THEN SOU
ND 3,200:GOTO 710
740 '-----
750 ' ALTAS DE FACTURAS
760 '-----
770 IF op$<>"A" THEN GOTO 1130
780 LOCATE #3,45,6:PRINT #3,USING "
###":nap
790 LOCATE #3,45,8:INPUT #3,"",A,B,
C
800 IF A>31 OR B>12 THEN SOUND 3,20
0:GOTO 790
810 LOCATE #3,45,8:PRINT #3,USING "
##/##/##":A,B,C
820 DIA$="00"+RIGHT$(STR$(A)),(LEN
(STR$(A))-1):me$="00"+RIGHT$(STR$(
B)),(LEN(STR$(B))-1):an$="00"+RIG
HT$(STR$(C)),(LEN(STR$(C))-1):dat
e$=RIGHT$(DIA$,2)+RIGHT$(me$,2)+RIG
HT$(an$,2)
830 LOCATE #3,46,10:INPUT #3,"",nfa
ct$:nfact$=UPPER$(nfact$)
840 IF LEN(nfact$)>7 THEN SOUND 3,2
00:GOTO 830
850 nfact$=" "+nfact$:nfac1$:
RIGHT$(nfact$,7)
860 LOCATE #3,46,12:INPUT #3,"",Imp
:
870 IF imp>9999999 THEN SOUND 3,200
:GOTO 860
880 LOCATE #3,45,12:PRINT #3,USING
"#####":imp
890 LOCATE #3,49,14:INPUT #3,"",iva
:
900 IF iva>99.9 THEN THEN SOUND 3,2
00:GOTO 890
910 LOCATE #3,49,14:PRINT #3,USING
"###.%":iva
920 totiv=ROUND(imp*iva/100)
930 LOCATE #3,45,16:PRINT #3,USING
"#####":totiv
940 LOCATE #5,30,2:INPUT #5,"CORREC
TO? (S/N): ",OPC$:OPC$=UPPER$(O
PC$)
950 CLS #5:IF OPC$="N" THEN 1100
960 IF OPC$="S" THEN 970 ELSE SOUND
3,200:GOTO 940
970 '-----
980 ' GRABA REGISTRO EN DISCO
990 '-----
1000 NAP$=" "+STR$(NAP):IMP$="
"+STR$(IMP):IVA$=" "+STR$(IV
A):
1010 NAP$=RIGHT$(NAP$,3):IMP$=RIGHT
$(IMP$,7):IVA$=RIGHT$(IVA$,4)
1020 IF OP=1 THEN REG$="A"+NAP$+NFA
CT$+IMP$+IVA$+DATE$+"S"
1030 IF OP=3 THEN REG$="A"+NAP$+NFA
CT$+IMP$+IVA$+DATE$+"R"
1040 !WRITE,& REG$,Nr,1
1050 IF op$<>"A" THEN 1100
1060 NAP=NAP+1:NR=NR+1:NR$="000"+MI
D$(STR$(NAP)),2,(LEN(STR$(NAP))-1)
:NR$=RIGHT$(NR$,3):ULAP=ULAP+1
1070 a$="*"+NR$
1080 !WRITE,& A$,1,1
1090 '*** FIN DE GRABACION
1100 LOCATE #5,29,2:INPUT #5,"MAS E
NTRADAS?(S/N): ",OPC$:OPC$=UPPER$(
OPC$)
1110 CLS #5:IF OPC$="N" THEN !CLOSE
:GOTO 160

```

```

1120 IF OPC$="S" THEN 610 ELSE SOU
ND 3,200:GOTO 1100
1130 '-----
1140 ' MODIFICACION O CONSULTA
1150 '-----
1160 LOCATE #3,45,6:INPUT #3,"",nap
:
1170 IF NAP>=ULAP THEN LOCATE #5,16
,2:INPUT #5,"Registro Consultado In
existente. Pulse <ENTER> ",pr$:CLS
#5:GOTO 1160
1180 IF NAP<>0 THEN 1320
1190 LOCATE #3,46,10:INPUT #3,"",nf
act$:nfact$=UPPER$(nfact$)
1200 IF LEN(nfact$)>7 THEN SOUND 3,
200:GOTO 1190
1210 nfact$=" "+nfact$:nfact$
=RIGHT$(nfact$,7)
1220 FOR X=2 TO ulap
1230 !READ,& A$,Y,1
1240 IF nfact$=MID$(A$,5,7) THEN GO
TO 1270
1250 NEXT
1260 LOCATE #5,2,7:INPUT #5,"No Hay
registrada ninguna factura con esa
clave. Pulse <ENTER> ",pr$:CLS #5:G
OTO 1100
1270 IF OP=1 AND RIGHT$(A$,1)="R" T
HEN 1250
1280 IF OP=3 AND RIGHT$(A$,1)="S" T
HEN 1250
1290 nap=x-1:NR=X
1300 LOCATE #3,45,6:PRINT #3,USING
"###":nap
1310 GOTO 1350
1320 nr=nap+1:!READ,& A$,Nr,1
1330 IF (RIGHT$(A$,1)="R" AND op=1)
THEN LOCATE #5,6,2:INPUT #5,"Regis
tro Consultado es I.V.A. REPERCUTID
O. Pulse <ENTER> ",pr$:CLS #5:GOTO
1130
1340 IF (RIGHT$(A$,1)="S" AND op=3)
THEN LOCATE #5,6,2:INPUT #5,"Regis
tro Consultado es I.V.A. SOPORTADO.
Pulse <ENTER> ",pr$:CLS #5:GOTO
1100
1350 IF LEFT$(A$,1)<>"A" THEN LOCAT
E #5,6,2:INPUT #5,"REGISTRO DADO DE
BAJA. Pulse <ENTER> ",pr$:CLS #5:GO
TO 1100
1360 date$=MID$(a$,23,6)
1370 NFACT$=MID$(a$,5,7)
1380 imp$=MID$(a$,12,7)
1390 iva$=MID$(a$,19,4)
1400 LOCATE #3,46,10:PRINT #3,nfact
$
1410 LOCATE #3,45,8:PRINT #3,LEFT$(
date$,2);"/";MID$(date$,3,2);"/";RI
GHT$(date$,2)
1420 LOCATE #3,45,12:PRINT #3,USING
"#####":VAL(imp$):
1430 LOCATE #3,49,14:PRINT #3,USING
"###.%":VAL(iva$):
1440 imp=VAL(imp$):iva=VAL(iva$)
1450 totiv=ROUND(imp*iva/100)
1460 LOCATE #3,45,16:PRINT #3,USING
"#####":totiv
1470 IF op$="C" THEN 1100
1480 IF op$="B" THEN 1540
1490 LOCATE #5,30,2:INPUT #5,"CORRE
CTO? (S/N): ",OPC$:OPC$=UPPER$(O
PC$)
1500 CLS #5:IF OPC$="N" THEN 790
1510 IF OPC$="S" THEN 1100 ELSE SOU
ND 3,200:GOTO 1490
1520 '-----
1530 ' BAJAS DE REGISTROS
1540 '-----
1550 A$="B"
1560 LOCATE #5,20,2:INPUT #5,"Confo
rme en darlo de baja? (S/N): ",OPC
$:OPC$=UPPER$(OPC$)
1570 IF OPC$="N" THEN 1610
1580 IF OPC$="S" THEN 1590 ELSE SOU
ND 3,200:GOTO 1560
1590 a$="B"
1600 !WRITE,& a$,nr,1

```



```

1610 :CLOSE:GOTO 190
1620 '-----
1630 'LISTADOS DE I.V.A.
1640 '-----
1650 CLS #1:CLS #3:CLS #5:TOTAL=0
1660 IF OP=2 THEN LOCATE #1,21,1:PR
INT #1," L I S T A D O      I.V.A.  S
O P O R T A D O "
1670 IF OP=4 THEN LOCATE #1,21,1:PR
INT #1," L I S T A D O      I.V.A.  R
E P E R C U T I D O "
1680 LOCATE #1,18,2:PRINT #1,"=====
=====
=====
1690 LOCATE #3,25,8: INPUT #3,"Fech
a inicio del listado D,M,A .. ",A
1,B1,C1
1700 IF A1=0 AND B1=0 AND C1=0 THEN
FL=1:GOTO 1810
1710 IF A1>31 OR B1>12 THEN SOUND 3
,200:GOTO 1690
1720 LOCATE #3,25,10: INPUT #3,"Fec
ha final del listado D,M,A .. ",
A2,B2,C2
1730 IF A2>31 OR B2>12 THEN SOUND 3
,200:GOTO 1720
1740 DIA1$="00"+RIGHT$(STR$(A1)), (
LEN(STR$(A1))-1):me1$="00"+RIGHT$(
STR$(B1)), (LEN(STR$(B1))-1):an1$=
"00"+RIGHT$(STR$(C1)), (LEN(STR$(C1
))-1):date1$=RIGHT$(an1$,2)+RIGHT$(
me1$,2)+RIGHT$(DIA1$,2)
1750 DIA2$="00"+RIGHT$(STR$(A2)), (
LEN(STR$(A2))-1):me2$="00"+RIGHT$(
STR$(B2)), (LEN(STR$(B2))-1):an2$=
"00"+RIGHT$(STR$(C2)), (LEN(STR$(C2
))-1):date2$=RIGHT$(an2$,2)+RIGHT$(
me2$,2)+RIGHT$(DIA2$,2)
1760 DIA1$=RIGHT$(DIA1$,2)+MID$(DIA
1$,3,2)+LEFT$(DIA1$,2)
1770 DIA2$=RIGHT$(DIA2$,2)+MID$(DIA
2$,3,2)+LEFT$(DIA2$,2)
1780 LOCATE #5,30,2: INPUT #5,"CORRE
CTO? (S/N) .. ",OPC$:OPC$=UPPER$(
OPC$)
1790 CLS #5:IF OPC$="N" THEN 1690
1800 IF OPC$="S" THEN 1810 ELSE SOU
ND 3,200:GOTO 1780
1810 LOCATE #5,25,2: INPUT #5,"DESEA
IMPRIMIR EL LISTADO? (S/N) .. ",PR
$:pr$=UPPER$(pr$)

```

```

1820 CLS #5 :CLS #3:IF pr$="S" THEN
PRINT #8,"":OPCI$=UPPER$(OPCI$)
1830 IF pr$="S" THEN PRINT #8,STRIN
G$(80,""):PRINT #8,CHR$(8E);"LISTA
DO DEL I.V.A. ":OPCI$;CHR$(814);CHR
$(8F);" ":a1:"/";b1:"/";c1:"->";a2
:"/";b2:"/";c2;CHR$(812):
1840 LOCATE #1,4,3:PRINT #1," AFUN
TE N.FACTURA FECHA
IMPORTE I.V.A. TOTAL PTS.
"
1850 IF PR$="S" THEN PRINT #8," A
PUNTE N.FACTURA FECHA I
MPORTE I.V.A. TOTAL PTS."
PRINT #8,STRING$(80,"="):
1860 REG$="IVADATOS":a$=""
1870 :OPEN,2 reg$,1,29,1
1880 :READ,2 a$,1,1
1890 c$=LEFT$(a$,1)
1900 nap$=MID$(a$,2,3):NR=VAL(nap$)
:nap=nr-1
1910 FOR CO=2 TO nr
1920 :READ,2 a$,CO,1
1930 IF OP=2 AND RIGHT$(A$,1)="R" T
HEN 1990
1940 IF OP=4 AND RIGHT$(A$,1)="S" T
HEN 1990
1950 IF LEFT$(a$,1)="B" THEN 1990
1960 FECH$=MID$(A$,23,6):FEC1$=RIGH
T$(FECH$,2)+MID$(FECH$,3,2)+LEFT$(F
ECH$,2)
1970 IF FL=1 THEN GOSUB 2020:GOTO
1990
1980 IF fec1$>= date1$ AND fec1$<=
date2$ THEN GOSUB 2020
1990 NEXT
2000 IF pr$="S" THEN PRINT #8:PRINT
#8,CHR$(8E);" Total I.V.A. List
ado -> ":USING "#####";total:PR
INT #8,STRING$(80,"="):
2010 LOCATE #5,31,2: INPUT #5,"FULS
E INTRO":IN$:CLS #1:CLS #5:CLS #3:
CLOSE:GOTO 250
2020 '*** Presentacion por pantalla
2030 NFACT$=MID$(A$,5,7)
2040 imp$=MID$(a$,12,7)
2050 lva$=MID$(a$,19,4)
2060 imp=VAL(imp$):iva=VAL(iva$)
2070 totiv=ROUND(imp*iva/100)
2080 PRINT #3,USING " ###

```

*Serie*  
**ORO**

```

% % #####, ##.#
#####,":(co-1):infact$:fe
ch$:imp:iva:totiv:
2090 TOTAL=TOTAL+TOTIV
2100 IF PR$="S" THEN PRINT #8,USING
" ##### %/8/8 #####
##, ##.# #####,":(co-
1):infact$:LEFT$(fech$,2):MID$(fech$
,3,2):RIGHT$(fech$,2):imp:iva:totiv
:
2110 RETURN
2120 '-----
2130 ' PROCESO DE ACUMULACION
2140 '-----
2150 CLS #3:CLS #5:CLS #1: control=
60:paq=1:reg=2:totalr=0:totals=0
2160 LOCATE #1,21,1:PRINT #1,"P R O
C E S O D E L I Q U I D A C I
O N"
2170 LOCATE #1,20,2:PRINT #1,"=====
=====
=====
2180 REG$="IVADATOS":A$=""
2190 :OPEN,2 reg$,1,29,1
2200 :READ,2 a$,1,1
2210 c$=LEFT$(a$,1)
2220 nap$=MID$(a$,2,3):nr=VAL(nap$)
:nr=nr+1:nap=nr-1:ulap=nr
2230 LOCATE #3,16,6: INPUT #3," Prim
er dia de liquidacion: (DD,MM,AA)..
":a1,h1,c1
2240 IF a1>31 OR b1>12 THEN SOUND 3
,200:GOTO 2230
2250 LOCATE #3,16,8: INPUT #3," Ulti
mo dia de liquidacion. (DD,MM,AA)..
":a2,b2,c2
2260 IF a2>31 OR b2>12 THEN SOUND 3
,200:GOTO 2250
2270 dia1$="00"+RIGHT$(STR$(A1)), (
LEN(STR$(A1))-1):me1$="00"+RIGHT$(
STR$(B1)), (LEN(STR$(B1))-1):an1$=
"00"+RIGHT$(STR$(C1)), (LEN(STR$(C1
))-1):dia1$=RIGHT$(an1$,2)+RIGHT$(
me1$,2)+RIGHT$(dia1$,2)
2280 dia2$="00"+RIGHT$(STR$(A2)), (
LEN(STR$(A2))-1):me2$="00"+RIGHT$(
STR$(B2)), (LEN(STR$(B2))-1):an2$=
"00"+RIGHT$(STR$(C2)), (LEN(STR$(C2
))-1):dia2$=RIGHT$(an2$,2)+RIGHT$(
me2$,2)+RIGHT$(dia2$,2)
2290 IF dia2$<=dia1$ THEN SOUND 3,2
00:CLS #3:GOTO 2230
2300 CLS #3:LOCATE #3,16,6:PRINT #3
," Primer dia de liquidacion: (DD,M
M,AA)..":RIGHT$(dia1$,2):"/":MID$(
dia1$,3,2):"/":LEFT$(dia1$,2)
2310 LOCATE #3,16,8:PRINT #3," Ulti
mo dia de liquidacion: (DD,MM,AA)..
":RIGHT$(dia2$,2):"/":MID$(dia2$,3
,2):"/":LEFT$(dia2$,2)
2320 LOCATE #5,30,2: INPUT #5,"CORR
ECTO? (S/N) .. ",OPC$:OPC$=UPPER$(
OPC$)
2330 CLS #5: IF OPC$="N" THEN CLS #
3:GOTO 2230
2340 IF OPC$="S" THEN 2350 ELSE SOU
ND 3,200:GOTO 2320
2350 LOCATE #5,25,2: INPUT #5," DES
EA UNA COPIA POR IMPRESORA? (S/N)..
": ,IM$: IM$=UPPER$(IM$)
2360 CLS #5:IF IM$<>"S" AND im$<>"N

```

## **TABLA DE SUBROUTINAS DEL PROGRAMA IVA**

10-150	Descripción del programa, sus límites y recordatorio de la necesidad de usar el programa RANDOM.BIN.
160-400	Menú principal del programa: definición de ventanas (líneas 200-240) y presentación de todas las opciones disponibles.
420-510	Registro IVA.
520-550	Inicialización del fichero «IVADATOS».
610-730	Pantalla principal.
750-960	Alta de facturas.
970-1120	Graba registro en disco.
1130-1510	Modificación y/o consulta de los datos.
1520-1610	Bajas de registros.
1620-2010	Listados de IVA.
2020-2110	Presentación por pantalla de los datos.
2120-2600	Proceso de acumulación.
2610-2660	Rutina de impresión.
2670-2790	Salto de página y cabecera (depende de la impresora).
2800-2820	Fin del programa.



# Serie ORO

```

" THEN SOUND 3,200:GOTO 2350
2370 IF reg=nr THEN 2550
2380 !READ, @ a$, reg, 1
2390 nap$= MID$(a$,2,3): nap= VAL(n
ap$):' * num. apunte
2400 nfac$=MID$(a$,5,7):' * num. fac
tura
2410 tipo$=RIGHT$(a$,1):' * tipo Sop
ortado o Repercutido
2420 est$=LEFT$(a$,1):' * estado Alt
a o Baja
2430 fech$=MID$(a$,23,6):dia$=MID$(
a$,27,2)+MID$(a$,25,2)+MID$(a$,23,2
)
2440 imp$=MID$(a$,12,7):imp=VAL(imp
$):' * importe de la factura
2450 iva$=MID$(a$,19,4):iva=VAL(iva
$):' * porcentaje I.V.A.
2460 totiv= ROUND(imp*iva/100):' * t
otal retenido
2470 IF est$="R" THEN 2540
2480 IF dia$>dia2$ THEN 2540
2490 IF dia1$>dia$ THEN 2540
2500 IF tipo$="S" THEN totals=total
s+totiv
2510 IF tipo$="R" THEN totalr=total
r+totiv
2520 IF im$="S" THEN GOSUB 2610:' *
subrutina impresora
2530 LOCATE #3,20,10:PRINT #3,"Subt
otalizando Registro No..: ";reg
2540 reg=reg+1:GOTO 2370
2550 CLS #3:LOCATE #3,16,6: PRINT #
3,"Total I.V.A. Soportado....: ";tot
als
2560 LOCATE #3,16,8: PRINT #3,"Tota
l I.V.A. Repercutido..: ";totalr
2570 IF im$="S" THEN PRINT #8:PRINT
#8,STRING$(80,"-"):PRINT #8,CHR$(&
E);CHR$(27);"-";CHR$(1);" TOTA
LES --->";CHR$(&14);CHR$(27);"-";CH
R$(&0);USING " #####
###, #####, "; totals; total
r:
2580 IF (im$="S" AND totals>totalr)
THEN PRINT #8,CHR$(27);CHR$(52);"
*** DIFERENCIA A SU FAVOR DE
";USING " #####, &"; (TOTALS-TOT
ALR);"Ftas.": PRINT #8,CHR$(27);CHR
$(53)
2590 IF (im$="S" AND totalr>totals)
THEN PRINT #8,CHR$(27);CHR$(52);"
*** DIFERENCIA EN SU CONTRA
DE";USING " #####, &"; (TOTALR-T
OTALS);"Ftas.": PRINT #8,CHR$(27);C
HR$(53)
2600 LOCATE #5,31,2:INPUT #5,"PULSE
ENTER ";in$:CLS #1:CLS #5:CLS #3:
CLOSE: GOTO 250
2610 '**** RUTINA DE IMPRESION
2620 IF control=60 THEN GOSUB 2670
2630 PRINT #8,USING " ### %
," Primer dia de liquidacion. (DD,MM
,AA)...: ";RIGHT$(dia1$,2);"/";MID$(
dia1$,3,2);"/";LEFT$(dia1$,2)

```

```

2310 LOCATE #3,16,8:PRINT #3," Ulti
mo dia de liquidacion. (DD,MM,AA)..
: ";RIGHT$(dia2$,2);"/";MID$(dia2$,3
,2);"/";LEFT$(dia2$,2)
2320 LOCATE #5,30,2: INPUT #5,"CORR
ECTO? (S/N)..: ",OPC$:OPC%=UPPER$(
OPC%)
2330 CLS #5: IF OPC$="N" THEN CLS #
3:GOTO 2230
2340 IF OPC$="S" THEN 2350 ELSE SOU
ND 3,200:GOTO 2320
2350 LOCATE #5,25,2: INPUT #5," DES
EA UNA COPIA POR IMPRESORA? (S/N)..
: ",IM$: IM%=UPPER$(IM%)
2360 CLS #5:IF IM$<>"S" AND im$<>"N
&/&/& #####, ##.# ";nap;nf
ac$;LEFT$(fech$,2);MID$(fech$,3,2);
RIGHT$(fech$,2);imp;iva;:
2640 IF tipo$="S" THEN PRINT #8,USI
NG " #####, ";totiv ELSE PRINT
#8,USING " #####
##, ";totiv
2650 control=control+1
2660 RETURN
2670 '*** SALTO PAGINA Y CABECERA
2680 PRINT #8,CHR$(27);"@" : '*** IN
ICIALIZA IMPRESORA
2690 PRINT #8:PRINT #8:PRINT #8,STR
ING$(80,"=")
2700 PRINT #8,CHR$(&E);" LIQUID
ACION I.V.A. ";CHR$(&14);CHR$(27);"
S";CHR$(0);CHR$(15);"Desde ";RIGHT$(
dia1$,2);"/";MID$(dia1$,3,2);"/";L
EFT$(dia1$,2);
2710 FOR x=1 TO 14:PRINT #8,CHR$(&8
);:NEXT x:'*** Retrocede la cabeza
de impresion
2720 PRINT #8,CHR$(27);"S";CHR$(1);
CHR$(15);"Hasta ";RIGHT$(dia2$,2);"
/";MID$(dia2$,3,2);"/";LEFT$(dia2$,
2);
2730 PRINT #8,CHR$(27);"H";CHR$(27)
;"T";CHR$(18)::'*** Codigos control
impresora
2740 PRINT #8," Pagina..: ";pag:
2750 PRINT #8," AFUNTE FACTURA
FECHA IMPORTE IVA S
OPORTADO REPERCUTIDO"
2760 PRINT #8,STRING$(80,"=");
2770 PAG=PAG+1:CONTROL=1
2780 RETURN
2790 '-----
2800 ' FIN DE PROGRAMA
2810 '-----
2820 !CLOSE:CLS:END

```



# RPA® Systems Inc

## PRESENTA

### CONTABILIDAD GENERAL 2 AMSTRAD 8256/6128

Programa de contabilidad de acuerdo con el plan general contable español.

Capacidad aproximada:

- Una unidad de disco: 500 cuentas, 2.000 asientos y movimientos ilimitados.
- Dos unidades: 700 cuentas, 3.000 asientos y movimientos ilimitados.
- Tres unidades: ilimitada.

#### Características generales:

- Compatibilidad con todas las impresoras del mercado que puedan trabajar con AMS-TRAD.
- Hasta 96 conceptos auxiliares creados por el usuario.
- Definición de la configuración elegida por el usuario (1, 2 ó 3 unidades de disco con sus respectivas capacidades de funcionamiento).

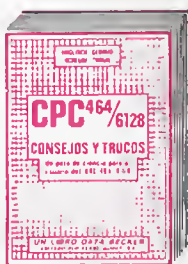
- Niveles, dígitos por nivel y cuenta programables por el usuario.
- Calculadora incorporada en el sistema sin salir de aplicación.
- En configuraciones ilimitadas no hay pérdida de apuntes contables.
- Pérdida mínima de datos ante cortes energéticos o desconexión involuntaria del ordenador.

#### SOLICITE NUESTRO CATALOGO

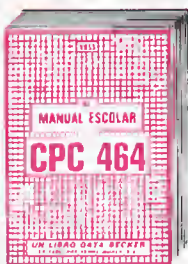
**24.500 pts. para 8256**  
**15.500 pts. para 6128** +IVA

**BAE TASA S.A.** - Galileo, 25 - 28015 Madrid - ☎ 447 97 51 ☎ 447 98 09

## DATA BECKER APUESTA FUERTE POR AMSTRAD



Ofrece una colección muy interesante de sugerencias, ideas y soluciones para la programación y utilización de su CPC-464. Desde la estructura del hardware, sistema de funcionamiento - Tokens Basic, dibujos con el joystick, aplicaciones de ventanas en pantalla y otros muchos interesantes programas como el procesamiento de datos, editor de sonidos, generador de caracteres, monitor de código máquina hasta listados de interesantes juegos.  
**CPC-464 Consejos y Trucos. 263 págs. P.V.P. 2.200,- ptas.**



Escrito para alumnos de los últimos cursos de EGB y de BUP, este libro contiene muchos programas para resolver problemas y de aprendizaje, descritos de una forma muy compleja y fácil de comprender. Teorema de Pitágoras, progresiones geométricas, escritura cifrada, crecimiento exponencial, verbos irregulares, igualdades cuadráticas, movimiento pendular, estructura de moléculas, cálculo de interés y muchas cosas más.  
**CPC-464 El libro del catálogo. 380 págs. P.V.P. 2.200,- ptas.**



PEEKs, POKEs y CALLs se utilizan para introducir al lector de una forma fácilmente accesible al sistema operativo y al lenguaje máquina del CPC. Proporciona además muchas e interesantes posibilidades de aplicación y programación de su CPC.  
**PEEKs y POKEs del CPC 464/6128. 180 pág. P.V.P. 1.600,- ptas.**



La técnica y programación del Procesador 280 son los temas de este libro. Es un libro de estudio y de consulta imprescindible para todos aquellos que poseen un Commodore 128, CPC, MSX u otros ordenadores que trabajen con el Procesador 280 y desean programar en lenguaje máquina.  
**El Procesador 280. 540 pág. P.V.P. 3.800,- ptas.**



EL LIBRO DEL FLOPPY del CPC lo explica todo sobre la programación con discos y la gestión relativa de ficheros mediante el floppy DDI-1 y la unidad de discos incorporada del CPC 664/6128. La presente obra, un auténtico estándar, representa una ayuda incomparable tanto para el que desee iniciarse en la programación con discos como para el más curtido programador de ensamblados. Especialmente interesante resulta el listado exhaustivamente comentado del DOS y los muchos programas de ejemplo, entre los que se incluye un completo paquete de gestión de ficheros.  
**El libro del Floppy del CPC. 353 pág. P.V.P. 2.800,- ptas.**



¡Dominar CP/M por fin! Desde explicaciones básicas para almacenar números, la protección contra la escritura, o ASCII, hasta la aplicación de programas auxiliares de CP/M, así como «CP/M interno» para avanzados, cada usuario del CPC rápidamente encontrará las ayudas e informaciones necesarias, para el trabajo con CP/M. Este libro tiene en cuenta las versiones CP/M 2.2, así como CP/M Plus (3.0), para el AMSTRAD CPC 464, CPC 664 y CPC 6128.  
**CP/M. El libro de ejercicios para CPC. 260 pág. P.V.P. 2.600,- ptas.**



**TEXTOMAT 8.800 ptas.**

¡El procesador de textos más vendido en Alemania, ahora también disponible para

**AMSTRAD**

BOLETIN DE PEDIDO

**FERRE - MORET S.A.**

Tuset n.º 8, entlo. 2.º Tel. 218 02 93  
BARCELONA 08006

Desee adj. n.º

Gastos envío: 300 ptas.

☐ Adjunto cheque

☐ Reembolso más gastos del mismo

NOMBRE

DIRECCION



## HACIA LA INTELIGENCIA ARTIFICIAL CON AMSTRAD.

Cuando uno se acerca a este libro, otro día por el título, espero encontrar en él, al menos, una aproximación a ese apasionante tema que sin duda alguna es la inteligencia artificial. La decepción de las que lleguen al libro malditas por este asunto no podrá ser mayor. Un curso de Basic incompleto es lo más que sacarán en claro. El libro comienza planteándose tres objetivos:

1. Mostrar diversas instrucciones del Basic.
2. Enseñar a escribir programas que trabajen con textos.
3. Enseñar a escribir programas que manipulen los textos que introduzco el usuario.

Con ciertas salvedades, el libro cubre esta pretensión. En la contrapartida se afirma que no se necesitan conocimientos de Basic para la lectura y comprensión del libro. Efectivamente así es, aunque más que no necesitarse, es completamente indispensable no poseer éstos, pues de no ser así, las 89 páginas, de las que consta, pueden convertirse en un auténtico suplicio por un lector más avanzado.

El programa central del libro es un intento de emulación del conocido programa Eliza, que simulando a un psicoanalista, va conversando con el usuario sobre su personalidad. Por desgracia este programa de nombre Sigmund, no ya sólo por estar escrito en Basic, lo cual supone un buen handicap para simular conversación, si no por falta de profundidad, se queda en una parodia.

Las últimas páginas del libro quieren canchalar la que en el mismo se denomina como curso de Choque de Basic. El cual más que de choque resulta chacante, pues dedican su totalidad al uso del print y del input. Toda esta nos lleva a que el título del libro muy bien podría haber sido, «Hacia el control aéreo con Amstrad» o «Hacia la parapsicología con Amstrad», con tal de haber cambiado un par de páginas por otros relacionados con estas temas. Al final el resulta-



AUTOR: JEREMY VINE.  
EDITORIAL: INDESCOMP, S.A.  
PAGINAS: 79.  
NIVEL: BASICO.  
PRECIO: 1.000 ptas.  
IDIOMA: CASTELLANO.

do habría sido el mismo una introducción al Basic del Amstrad.

Poco más puede decirse de este pequeño libro, un título excesivamente pretencioso que atrae a lectores con un nivel de Basic elevado, y que aleja a lectores que empiezan en la programación y encontrarán en él una guía para sus primeros pasos.

## CODIGO MAQUINA PARA PRINCIPIANTES CON AMSTRAD.

Yo seducido por el mundo de los videojuegos, movido por el ánimo de lograr mayores velocidades y un aprovechamiento máximo de la memoria o simplemente el afán de entender mejor la máquina por dentro, de lo que no cabe duda es de que el programador Basic, poseedor de un ordenador doméstico, acabará siendo atraído hacia el mundo del código máquina. El libro de Kramer intenta acercar este nuevo medio al nutrida grupa de usuarios de Amstrad que apasionadas por este tema, se encuentran desorientados.

Aunque el título del libro pudiera sugerir simplicidad, nada más lejos de la realidad. A pesar de la sencillez de su lectura, constituye por sí sola, un completo manual del lenguaje materno del Z80. A pesar de que el libro contenga algunos ejemplos comparativos Basic/Ensamblador, no hay ninguna razón de peso que pueda hacer pensar que para la lectura del libro sea necesaria un dominio del Basic.

Se echa en falta en un pequeño manual realizada en Basic, que aunque sencillo, permitiese al principiante comenzar o manejar este tipo de programas, así como poder escribir sus propias rutinas, en decimal o hexadecimal permitiéndole ubicarlas a placer en la memoria. No obstante, al final del libro se encuentran un cargador Hexadecimal que intenta cubrir esta ausencia, lo que unido al hecho de que todos los programas vienen desensamblados facilita notablemente la tarea.

El libro cubre desde las nociones más elementales sobre registros hasta una descripción,

quizá algo sumera de tratamiento de las interrupciones del Z80 haciendo hincapié en las que el Amstrad puede manejar con más eficacia. A lo correcto utilización y descripción de lo pila, se le dedica un capítulo completo en el cual se explica con todo lujo de detalles los beneficios de su utilización y los errores que pueden surgir debidos a una mala utilización. Una parte del libro se dedica a describir sin excesiva profundidad las comunicaciones que podemos realizar con el exterior o través del bus de direcciones cuya comprensión nos permite poner a nuestro Amstrad en contacto con multitud de periféricos, temo absolutamente prohibitivo por poder ser tratado desde el Basic.



El último capítulo del libro, el quince, intenta explicar cómo el programador de ensamblar puede sacar partida al uso del sistema operativo, numerosas rutinas que nuestro CPC conoce y utiliza puede compartirlas con nosotros siempre y cuando, naturalmente, conozcamos tanta su funcionamiento, como su ubicación. El apéndice G del libro describe un buen número de éstas, unas treinta, que servirán de gran ayuda cuando nos decidamos a elaborar nuestros primeros programas.

Sin duda alguno, el libro de Kramer constituirá un primer paso hacia el dominio de este apasionante tema que es el código máquina, para todos aquellos usuarios de la familia CPC de Amstrad.

AUTOR: STEVE KRAMER.  
PAGINAS: 175.  
EDITORIAL: INDESCOMP, S.A.  
PRECIO: 1.900 ptas.  
NIVEL: MEDIO  
IDIOMA: CASTELLANO.



## CPC-464 FIRMWARE.



Si hubiese que confeccionar una lista de todos los libros útiles al programador de cualquiera de los CPC, o lo coheza de ésta se encontraría sin género de dudas, el manual del firmware. Brillante, pues no existe otro adjetivo, ha sido la publicación de este tomo donde se recoge, con todo lujo de detalles, una buena parte del sistema operativo. El celo, pues otra razón no se concibe, con que otros diseñadores han construido sus máquinas, ha impedido al usuario utilizar, compartir, el sistema operativo con la máquina. No ha sido así en el caso que nos ocupa, lo que unido a la facilidad de acceso que la ubicación en RAM de un bloque de soltos, ha hecho posible la existencia de este libro.

Las posibilidades que el conocimiento de este trozo del sistema operativo conlleva son más que infinitos, no ya sólo al programador en Basic, quien podrá ver aumentada notablemente la velocidad de sus programas, sino para el programador en máquina quien se encontrará con una librería de rutinas diseñadas para obtener un máximo rendimiento. Sin embargo, a parte de este último grupo, es de destacar el uso que los aficionados o programar en otros lenguajes, que si bien más potentes que el Basic, no disponen de todas esas maneras que el Basic nos ofrece, ventanas, colores, ubicación del cursor en la pantalla, gráficos y un sinfín de etcéteras, después de un poco de práctica, serán tan sencillos de utilizar como lo eran desde el Basic.

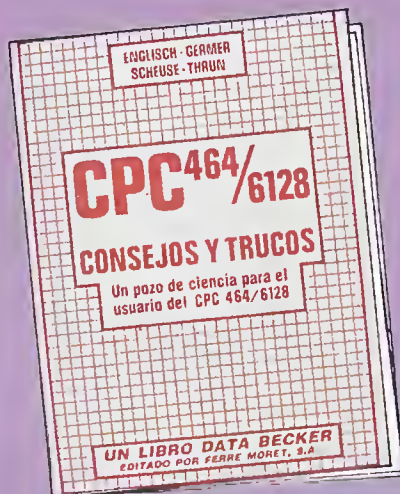
Aunque el nombre del libro sea manual del firmware, también se contemplan en él algunos interesantes aspectos de los distintos elementos que conforman el Hardware de los CPC, explicando cómo podemos comunicarnos con ellos a través de los distintos «ports».

El libro, razón desconocida, sólo tiene traducido al castellano la portada, esperamos, ya que el precio del libro así lo merece, aparezca pronto la versión traducida. Por el momento, sin embargo, habrá que conformarse con leerlo diccionario en mono.

AUTOR: **BRUCE GODDEN.**  
EDITORIAL: **AMSOFT.**  
PAGINAS: **393**  
PRECIO: **3.400 ptas.**  
NIVEL: **AVANZADO.**  
IDIOMA: **INGLES.**

Esperemos que esta iniciativa se expanda y alcance a las nuevas rutinas que se han incorporado en el 6128, particularmente las que troton del disco y pronto Amsoft nos deleite con la publicación de un anexo que recoja éstos. Por cierto aunque en el libro no se mencione la «compatibilidad» con el 664 y con el 6128 está asegurada.

## CPC 464/6128 CONSEJOS Y TRUCOS.



Gráficos, sonido, lenguaje máquina, memoria de programa, rutinas de utilidad y algunos programas de aplicación, son las partes que dividen las doscientas setenta páginas de que consta el libro de Data Becker.

Destinado a conocedores del Basic, el libro puede constituir un verdadero hallazgo para los que ya han practicado algún dialecto del Basic en otras máquinas, pues la atención de este libro se centra de forma casi exclusiva, en las instrucciones que más se alejan del Basic standard y que constituyen en muchas ocasiones la piedra angular del éxito en la programación de cualquiera de los CPC.

Como ya anunciábamos, el capítulo primero se dedica al estudio de los gráficos, término excesivamente estricto pues en sus páginas se recogen, junto al estudio de los comandos gráficos, los conocimientos necesarios para el tratamiento de ventanas de texto, la ubicación de la información en la pantalla y en fin todo lo concerniente a la utilización de la pantalla.

# Libros

El capítulo dos emplea todo su extensión en el trombamiento que con el **Amstrad** podemos realizar del sonido. Lo primero que se intenta es crear en el lector una base teórica de los parámetros que constituyen la definición de un sonido determinado. Herramienta indispensable para los que deseen un completo dominio de las extraordinarias posibilidades sonoras del **Amstrad**.

En el siguiente capítulo se pretende aunque de forma muy somera dar una visión muy general de lo que es el lenguaje máquina y la manera de su utilización. Da una perspectiva muy superficial del tema, aunque en ocasiones se olvida la personalidad del lector potencial de este libro, elevándose el nivel hasta grados de una más que dificultosa comprensión.

La última parte teórica del libro lo constituye un capítulo desde el que se intenta desvelar la forma en que nuestro **Amstrad** olmo-ceno los programas y datos en los 64 ó 128 K de memoria central.

Los dos últimos capítulos se han dispuesto de una forma eminente práctica, el primero acoge en sus páginas algunas rutinas útiles para el usuario, entre los que son de destacar especialmente dos, uno que permite la impresión de fondo y otra que permite comunicar nuestro **Amstrad**, vía cable, con un Commodore 64. Las páginas finales del libro comprenden una descripción detallada de cuatro programas, dos de utilidad, bastante buenos y dos juegos que no son capaces de salir de la mediocridad general.

Aunque lejos de la línea elegida en el libro anterior de Becker, este libro puede constituir una buena alternativa para los que habiendo leído el manual escolar no calmaron su sed de Basic.

Sólo se hecha en falta el hecho de que se hubiera realizado una verdadera revisión del libro, tras la aparición del 6128. Naturalmente todo lo que en el libro se dice es válido para éste, sin embargo, se deberían haber incluido las nuevas prestaciones que este ordenador trajo a la familia CPC. Por el contrario el editor se ha conformado por añadir **16128**, allí donde ponía 464.

AUTOR: **ENGLISH-GERMER-SCHUESE-TRHUN.**  
PAGINAS: **271.**  
EDITORIAL: **DATA BECKER.**  
PRECIO: **2.200 ptas.**  
NIVEL: **MEDIO.**  
IDIOMA: **CASTELLANO.**



## TECNICAS DE PROGRAMACION AVANZADA CON AMSTRAD.



Es difícil imaginar la intención del autor cuando se planteó la idea de escribir este libro. Para describir sus cien primeras páginas, sería necesaria realizar una descripción detallada de cada una de ellas para lograr hacer llegar al interesado, el espíritu del libro. En este primer bloque se recogen un buen número de interesantes resultados y que, sin embargo, fácilmente pueden perderse por la falta de metadatos del libro. Una considerable carga de buena voluntad, será necesaria por parte del lector para poder sacarle su máxima rentabilidad.

No debe ser esta interpretación de forma estrictamente negativa, es interesante poder encontrar en el mercado diversas alternativas al aprendizaje del Basic y ésta es, sin duda, tan buena como cualquiera. El lector curioso y deseoso de avanzar por sí mismo encontrará en este libro las pistas necesarias que le conducirán a este objetivo.

Aritmética binaria, almacenamiento interno de cadenas alfanuméricas y números, manipulación de bloques de memoria, direcciones reales de variable, son, entre otras, algunas de las temas que a lo largo del libro se tra-

tan. Sería difícil precisar si es como consecuencia de una mala traducción del libro a un lenguaje informático algo extravagante por parte del autor, pero el caso es que las técnicas que en el libro se manejan pueden resultar en ocasiones chocantes, incluso para el lector más avezado.

La segunda parte del libro es mucho más ordenada en ella se trata, primera y de forma muy rápida, una breve descripción de qué es y para qué sirve el sistema operativo; y en una segunda parte se implementan, sin entrar en muchas explicaciones, un juego de nuevas instrucciones que vienen a suplir las ausencias importantes de Locomotive. El tratamiento del sonido por medio de notas puras y la utilización de sprites. En total seis nuevas comandos que naturalmente podrán ser usados en nuestros programas como si de verdaderas instrucciones Basic se trataran.

En resumen se trata de un libro interesante, de cuyas páginas podrán extraerse conocimientos importantes. Un libro que exige mucha y del que, particularmente, recomendaría no pasar más de quince páginas al mes. Tres días para leer y veintisiete para investigar.

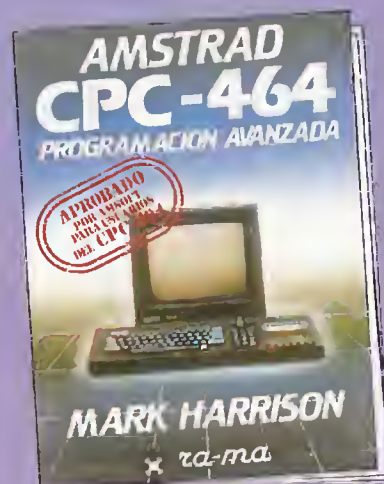
### CPC-464 PROGRAMACION AVANZADA.

Con un título prácticamente indistinguible al anteriormente comentado de la misma editorial, se nos presenta ahora un libro de concepción totalmente distinta. Tanto en las abjetivas como en la forma de conseguir las cosas las libras no tienen nada en común salvo hablar del **Amstrad**. En este caso, verdaderamente se estudia y describe una programación más racional y ordenada.

Los temas que el pequeño libro describe son muchas y variadas abarcando desde el tratamiento de cadenas hasta el dominio del sonido.

Un capítulo especialmente interesante, lo constituye el número siete que intenta recrearse en la implementación de estructuras de datos. Listas, pilas, grafos y árboles son tratados con elegancia, no tanto ya en la explicación como en los ejemplos que se utilizan para su descripción. Estructuras que si bien pudieran aparecer como cosas patológicas desde nuestra perspectiva Basic permiten, no obstante, resolver con sencillez diversos problemas y la que es más importante pueden constituir de marca perfecta como introducción a la utilización de la memoria dinámica, que lenguajes más potentes, como puede ser el Pascal, contemplan.

Es importante y destacable el hecho de que



el libro no se contente con mostrar al lector la utilización de un conjunto, de instrucciones más o menos abigarradas del Basic. Independientemente de esta, que también se trata, se ofrecen programas, ya clásicos, que utilizan potentes algoritmos para la ordenación de datos. Burbuja, cancha y Quick sort son, entre otros, algunas de las algoritmos que el lector podrá escoger para ordenar los datos que su programa maneje.

Queda un poco de la información que de archivos en memoria externa se da. Rasga común a todas las libras que el mercado ofrece. No se alcanza a comprender cómo un tema al que tanta partida se le puede extraer, permanezca aún en la más oscura de las sombras. ¿Quién sabe quizá alguien se anime?

Por último, decir que el título CPC-464, no debe confundir al usuario del 6128. El libro es perfectamente válido para su máquina. Por cierto, no estaría demás una reedición del libro considerando las nuevas posibilidades que el 128 ofrece.

Nota al traductor: Las palabras microprocesador y microordenador expresan dos conceptos totalmente distintos, razón por la cual no entienda cómo ha podido confundirlas a lo largo de toda el libro.

AUTOR: **KEITH HOOK.**  
EDITORIAL: **RA-MA.**  
PAGINAS: **161.**  
NIVEL: **AVANZADO.**  
PRECIO: **1.600 ptas.**  
IDIOMA: **CASTELLANO.**

AUTOR: **MARK HARRISON.**  
EDITORIAL: **RA-MA.**  
PRECIO: **1.400 ptas.**  
PAGINAS: **143.**  
NIVEL: **AVANZADO.**  
IDIOMA: **CASTELLANO.**



## MUSICA Y SONIDOS CON AMSTRAD.

Música y sonidos con **Amstrad** constituye, por desgracia, un libro con idéntico trayectoria al anteriormente comentado del mismo autor, «Hacia la inteligencia artificial con **Amstrad**». Un excesivo sensacionalismo, beneficioso en pequeñas dosis para lograr atraer la atención del lector, se convierte aquí en protagonista, con la indefectible profunda decepción tras el consabido RUN. Si nadie espera milagros, ¿por qué ofrecerlos?

El libro comienza exponiendo los característicos claves del sonido, tono, volumen y duración, junto a la utilización de los tres distintos canales, más el de ruido, con los que cuenta el **Amstrad**. Posteriormente se explica cómo estos parámetros se controlan mediante el orden sound.



Los envoltorios de tono y volumen controlados por ENT y ENV respectivamente, se explican después de haber realizado algunos ejemplos básicos. Se ha incluido también en el libro un par de capítulos que pretenden dar al lector unos conceptos muy básicos sobre música. En él se tratan los pentagramas, las notas, el compás y el ritmo.

El capítulo ocho se dedica a la creación de efectos especiales, en él se exponen algunos ejemplos, más o menos originales, como simulación de disparos láser, caídas de bombas, estallidos y sirenas, resueltos con más o menos éxito por parte del autor.

En el capítulo ocho nos encontramos con la terrible decepción. El sintetizador de sonido que se promete en la portada del libro, se convierte en un insulso programa emisor de pitidos. Aunque naturalmente qué otro cosa puede esperarse de un programa de exactamente ocho líneas en su versión reducida y coloreada en el omplido.

El último capítulo, dos páginas, vuelve a ho-

AUTOR: **JEREMY VINE.**  
PAGINAS: **94.**  
EDITORIAL: **INDESCOMP, S.A.**  
PRECIO: **1.300 ptas.**  
NIVEL: **BÁSICO.**  
IDIOMA: **CASTELLANO.**

cer hincapié en el tema de los efectos especiales, tres programitos de cuatro líneas cada uno.

El opéndice F del libro puede resultar interesante para los que deseen profundizar en el AY-3-8912, circuito integrado que tiene como misión generar el sonido en el **Amstrad** y de cuyo conocimiento podrán, sin duda, sacarse algunas posibilidades innecesables desde el Basic. Asimismo se ofrecen las direcciones del sistema operativo relacionados con el sonido. Sin embargo, el hecho de no omitir éstos de la información indispensable sobre los registros, los hacen totalmente insertables para el usuario.

Como colofón bien pudieran decirse: mucho ruido y pocas nueces.

## PROGRAMACION CON AMSTRAD.

Resultado curioso la incursión de Ian Sinclair, en el mundo editorial para mostrar el funcionamiento de un ordenador que no pertenece a sus dominios, lo digo ZX, con sus modelos 81 y Spectrum en todas sus versiones así como del QL.

El libro constituye un completo manual para el principiante, sin entrar en pormenores excesivos, en él se recoge la práctica totalidad del conjunto de instrucciones que conforman el Basic Locomotive. Sin excesivos originalidades, el libro comienza verdaderamente desde cero para alcanzar un nivel medio de conocimientos al final de su lectura. Conceptos de programación del teclado, detección de errores por el sistema y todo un capítulo de más de veinte páginas sobre el sonido, que permitirán al lector un perfecto control del mismo, avolan esto afirmación.

De forma paralela al desarrollo del texto se insertan un buen número de programas que permiten al lector ir leyendo o a la vez que compruebo los resultados predichos y hacen, que dudo cabe, mucho más entretenido la lectura.

Un dato especialmente interesante y digno

## Libros



de mención lo constituye el contenido del capítulo siete, en cuyos páginas se explica el manejo de ficheros en cinta. Si bien, lo verdaderamente productivo sería su aplicación al disco, no es de despreciar esta información, por otro parte, perfectamente utilizable por la unidad de disco.

El libro constituye por sí sólo, uno bueno alternativo al manual que lo acompaño con el ordenador y que si bien se alcanza una calidad nada desdeñable, puede resultar, sin embargo, árido para los que comienzan en la informática. En sus páginas se recogen incluso datos referentes a la instalación, cuestión quizá excesiva si tenemos en cuenta que ésta viene perfectamente detallada en los primeros páginas del manual oficial.

No obstante bienvenido sea, este libro auto-suficiente y que permitirá al principiante aguar durante algún tiempo para leer el manual y socorle de esto forma su máximo rendimiento, tras la lectura de este libro que nos llega de los monjes de Sinclair.

AUTOR: **IAN SINCLAIR.**  
EDITORIAL: **INDESCOMP, S.A.**  
PAGINAS: **180.**  
PRECIO: **2.100 ptas.**  
NIVEL: **BÁSICO.**  
IDIOMA: **CASTELLANO.**



# ESPECIAL RSX

Por Alberto Sñer

**Todos los poseedores de ordenadores Amstrad conocemos la gran potencia de su sistema operativo, y sabemos que éste nos ofrece una serie de posibilidades que otros ordenadores no pueden facilitar.**



Una de estas ventajas se encuentra en las extensiones residentes del sistema, a para entendernos mejor, las RSX. Esta nos permite definir nuevas camandas que podrán ser utilizados desde el Basic.

Dada pues la importancia de las RSX, hemos creído conveniente recopilar todos los nuevos comandos definidos hasta el momento, en un solo programa, para que puedan ser utilizados conjuntamente, añadiendo además varios comandos nuevos que esperamos sean de vuestro interés.

A continuación repasaremos todas las nuevas camandas e intentaremos explicar en lo posible su funcionamiento y la forma en que deben ser llamados desde Basic.

En primer lugar nos encontramos con un comando cuya función es resetear el ordenador, para reteniendo el programa Basic que tenemos en memoria. No necesita ningún parámetro, y deberemos llamarlo de la siguiente forma:

## I RESET

Describiremos las llamadas al firmware e intentaremos explicar la función de cada una de ellas:

#B906	Habilita la ROM Inferior.
#BD37	Reinicializa la tabla de saltos del sistema.
#BB00	Inicializa el buffer de teclado.
#BBFF	Inicializa la pantalla, dejando las tintas y el modo a sus valores por defecto.
#BC65	Inicializa Cassette.
#BCA7	Resetea la cola de sonido.
#BB4E	Inicializa la pantalla de texto.
#B90C	Restablece el previo estado de la ROM dado por el valor que contiene el acumulador.

Así pues, el comando RESET se basa en la utilización de las anteriores llamadas al firmware, y una vez realizadas vuelve al Basic.

Otra de las nuevas camandas introducidas, deberemos llamarlo desde Basic de la forma que a continuación se indica:

## I SETCLOCK, bajo, alto

su función es inicializar el reloj interno del ordenador. Si se utiliza sin ningún parámetro, pone el reloj a cero, si es utilizada con el parámetro 'baja', el valor de dicho parámetro es colocado en los dos bytes menos significativos del contador del reloj, y se utiliza además el parámetro alto, ese valor es colocada en los dos bytes más significativos del contador del reloj.

Utiliza una sola llamada al firmware, que es la siguiente:

#BD10	Inicializa el contador del reloj con los valores que contienen los registros dobles DE y HL. D contiene el byte más significativo y L el menos significativo.
-------	---

J. Siemens





Por lo tanto, si nuestra llamada a esta rutina no contiene ningún parámetro, los registrados DE y HL contendrán el valor cero por lo que el contador de reloj se inicializará a cero. Si dicha llamada contiene un parámetro, ésta se pasará al registro HL, así DE contendrá cero, por lo que los dos bytes más significativos del contador serán cero y los dos bytes menos significativos contendrán el valor dado por nosotros. Si la llamada contiene dos parámetros, entonces el segundo parámetro lo colocaremos en DE, por lo que modificaremos los cuatro bytes del contador del reloj.

Otro de los nuevos comandos, debe llamarse de la forma que se indica:

**I GPEN,opcion,color**

utiliza dos llamadas al firmware:

#BBDE	Coloca lo tinto poro gráficos. El acumulador debe contener el valor de lo tinto.
#BB5A	Imprime un carácter en pontollo u obedece un carácter de control. El acumulador debe contener el valor de dicho carácter.

La rutina utiliza la llamada a la dirección #BB5A para poner el modo de impresión dado por el parámetro opción. Así pues si este parámetro es 0, la impresión será en modo normal, si es 1, la impresión se realizará en modo XOR, si vale 2, se realizará en modo AND, y si es 3 en modo OR.

El próximo comando con el que nos encontremos debe ser llamado como sigue:

**GPAPER,color**

asigna el color del papel poro gráficos, el parámetro 'color' debe contener ese valor. Utiliza uno solo llamado al firmware:

#BBE4	Asigno el color del papel dado por el valor que contiene el acumulador.
-------	---

Por lo tanto, al llamar a dicha rutina se pasa al registro A el valor que contiene el parámetro 'color' y se llama a dicha rutina del firmware.

Otro de los nuevos comandos es:

**I GET, X@X%**

donde X puede ser cualquier variable entera. Utiliza la siguiente llamada:

#BB06	Espero o que se pulse un carácter desde el teclado y pongo su código ASCII al acumulador.
-------	---

Al utilizar este comando, espera a que se pulse una tecla y deja su código ASCII en la variable entera X%.

Por lo tanto, cuando pulsemos una tecla, la rutina colocará el código de la misma en la dirección donde se encuentra el contenido de la variable entera. Otro de los nuevos RSX es el que se describe a continuación:

**I FLUSH,buffer**

el parámetro 'buffer' puede ser 0 ó 1. Si es cero vacía el buffer de teclado si vale vacío el buffer de sonido. Utiliza dos llamadas al firmware:

#BCA7	Vacío el buffer de sonido.
#BB09	Vacío el buffer de teclado.

Así pues la rutina chequea el parámetro dado, si éste vale 1 llama a la primera rutina y si es cero llama a la segunda rutina del firmware.

A continuación nos encontramos con:

**I FILL,x,y,c**

los parámetros 'x' e 'y' son las coordenadas de un punto dentro del recinto a plotear y 'c' es el color de la tinta con la que se va a rellenar.

Utiliza las siguientes rutinas del firmware:

#BBDE	Selecciono pluma poro gráficos.
#BBF0	Devuelve en el acumulador el color de lo tinto que se encuentra en los coordenados dados por HL y DE.
#BBE4	Selecciona el papel poro gráficos.
#BC11	Devuelve en el acumulador el modo de pontollo.
#BBC0	Ploteo en los coordenados dados por HL y DE.
#BBF6	Dibuja una línea en las posiciones dados por DE y HL.

Los dos comandos que tenemos a continuación son los siguientes:

**PRINT.UP,@A\$**

**I PRINT.DOWN,@A\$**

Estos nos imprimen un carácter en pantalla girado noventa grados a la derecha o a la izquierda.

Dado que los funcionamientos de estos dos RSX fueron tratando exhaustivamente en su día, pasaremos a los nuevos comandos incluidos en este artículo.



El primero de ellos debe ser llamado desde Basic, de la siguiente forma:

DSCROLL, ancho, alto, x, y

Produce un scroll a la derecha de un área de pantalla definida por los parámetros anteriores. El parámetro 'ancho' nos da la anchura de esa zona de pantalla, 'alto' define la altura del área de pantalla, y 'x' e 'y' son las coordenadas correspondientes a la esquina superior izquierda del bloque de pantalla que queremos 'scrolear'.

Únicamente utiliza una llamada al firmware que es la siguiente:

#BC1A      Calcula la dirección de pantalla correspondiente a las coordenadas dadas en el registro HL.

El scroll realizado por esta rutina puede ser de mucha utilidad para presentaciones de programas o bien para la realización de programas de juegos, ya que el área de pantalla a tratar puede ser definida por nosotros en cada momento.

La última rutina incluida en este programa especial de extensiones residentes del sistema, deberá ser llamada de la siguiente forma:

I SCROLL, ancho, alto, x, y

Es exactamente igual que la descrita anteriormente, e incluso utiliza la misma llamada al firmware que la anterior. La única diferencia entre ellas es que la primera realiza un scroll a la derecha de una zona de pantalla y esta última realiza un scroll a la izquierda de esa zona de pantalla.

Esperamos que estas rutinas os sean útiles para la realización de vuestros propios juegos, ya que ellas solas por sí mismas constituyen pequeños bloques de programas que por sí mismos nos son nada, pero que unidos convenientemente pueden llegar a formar un programa de juegos o utilidades con bonitos efectos.



## PROGRAMA 1

```
10 FOR N=9000 TO 9300
20 READ A:SUMA=SUMA+A
30 POKE N,A
40 NEXT
50 IF SUMA<5000 THEN PRINT "ERROR
EN DATAS"
60 DATA 1,9,144,33,176,144,195
70 DATA 209,188,59,144,195,17,145
80 DATA 195,204,144,195,241,144,195
90 DATA 7,145,195,46,145,195,64
100 DATA 145,195,81,145,195,242,145
110 DATA 195,251,145,195,106,146,19
5
120 DATA 128,146,0,0,0,0
130 DATA 0,0,0,0,0,0
140 DATA 0,0,0,82,69,83,69
150 DATA 212,83,69,84,67,76,79
160 DATA 67,203,71,80,69,206,71
170 DATA 80,65,80,69,210,71,69
180 DATA 212,76,76,85,83,200,70
190 DATA 73,76,204,80,82,73,78
200 DATA 84,46,68,79,87,206,80
210 DATA 82,73,78,84,46,85,200
220 DATA 58,83,67,82,79,76,204
230 DATA 73,83,67,82,79,76,204
240 DATA 0,0,0,0,0,0
250 DATA 0,0,0,0,0,0
260 DATA 0,0,0,0,0,0
270 DATA 0,0,0,0,0,0
280 DATA 0,0,0,0,0,0
290 DATA 0,0,0,0,0,0
300 DATA 0,0,0,0,0,0
310 DATA 0,0,0,0,0,33,192
320 DATA 144,126,205,90,187,35,167
330 DATA 32,240,201,82,83,88,32
340 DATA 181,114,114,111,114,13,10
350 DATA 0,33,0,17,0,0
360 DATA 167,202,16,189,61,32,9
370 DATA 221,110,0,221,102,1,195
380 DATA 16,189,221,94,0,221,84
390 DATA 1,221,110,2,221,102,3
400 DATA 195,16,189,254,2,194,180
410 DATA 144,221,126,0,205,222,187
420 DATA 62,23,205,90,187,221,126
430 DATA 2,195,90,187,61,194,100
440 DATA 144,221,126,0,195,228,187
450 DATA 205,6,185,245,205,55,189
460 DATA 205,6,187,205,255,187,205
470 DATA 181,188,205,167,188,205,90
480 DATA 187,205,78,187,241,195,12
490 DATA 185,61,194,180,144,205,6
500 DATA 187,221,110,0,221,102,1
510 DATA 119,35,54,0,201,61,194
520 DATA 180,144,221,126,0,167,194
530 DATA 167,188,205,9,187,56,251
540 DATA 201,254,3,194,180,144,221
550 DATA 126,0,205,222,187,221,110
560 DATA 2,221,102,3,221,94,4
570 DATA 221,06,5,213,237,83,199
580 DATA 129,34,201,129,205,240,187
590 DATA 58,203,129,60,205,228,187
600 DATA 205,17,188,62,4,56,6
610 DATA 62,6,12,62,5,5
620 DATA 204,129,205,192,145,237,46
630 DATA 205,163,145,40,247,201,34
640 DATA 83,199,129,205,192,145,9
650 DATA 205,163,145,40,247,201,34
660 DATA 199,129,235,42,201,129,205
670 DATA 240,187,33,203,129,190,201
680 DATA 237,91,199,129,229,205,240
690 DATA 187,33,203,129,190,225,201
700 DATA 42,201,129,35,205,178
710 DATA 145,40,249,43,43,229,42
720 DATA 201,129,43,43,205,178,145
730 DATA 40,249,35,55,237,91,190
740 DATA 129,205,192,145,235,237,91
750 DATA 199,129,205,247,187,42,199
760 DATA 129,175,237,75,204,129,71
770 DATA 201,254,1,194,180,144,62
780 DATA 10,24,7,254,1,194,180
790 DATA 144,62,11,50,150,160,221
800 DATA 110,0,221,102,1,70,35
810 DATA 94,35,86,235,126,229,197
820 DATA 205,165,187,205,6,185,235
830 DATA 14,0,58,150,160,245,254
840 DATA 18,204,72,146,196,89,146
850 DATA 205,9,185,62,255,33,151
860 DATA 160,205,168,187,62,255,205
870 DATA 90,187,241,205,90,187,62
880 DATA 8,205,90,187,193,225,35
890 DATA 16,202,201,33,151,160,26
900 DATA 6,8,23,203,30,35,16
910 DATA 250,19,13,32,240,201,33
920 DATA 159,160,26,6,8,43,23
930 DATA 203,22,16,250,19,13,32
940 DATA 240,201,254,4,194,180,144
950 DATA 62,1,50,225,146,205,134
960 DATA 146,201,254,4,194,180,144
970 DATA 62,0,50,225,146,205,134
980 DATA 146,201,221,126,6,50,224
990 DATA 146,33,0,8,22,0,221
1000 DATA 94,6,237,82,34,228,146
1010 DATA 33,0,8,25,34,226,146
1020 DATA 221,110,0,221,102,2,58
1030 DATA 225,146,254,0,32,7,221
1040 DATA 126,4,205,187,146,201,221
1050 DATA 126,4,205,230,146,201,245
1060 DATA 35,229,205,26,188,6,8
1070 DATA 197,229,209,35,6,0,58
1080 DATA 224,146,61,79,26,237,176
1090 DATA 18,237,75,228,146,9,193
1100 DATA 16,233,225,24,61,32,220
1110 DATA 201,0,0,0,0,0,0
1120 DATA 245,35,229,205,26,188,6
1130 DATA 8,197,229,209,43,6,0
1140 DATA 58,224,146,61,79,26,237
1150 DATA 184,18,237,75,226,146,9
1160 DATA 193,16,233,225,241,61,32
1170 DATA 220,201,0,0,0,0,0
```

## LISTADO 1

```
VERA 4189 8 10 R R 00
9000 2180 7 20 I D 00
9003 2180 7 20 I D HL,0
9006 2180 7 20 P HRCI 1
9009 2180 7 20 P HRCI 1
9012 2180 7 20 P HRCI 1
9015 2180 7 20 P HRCI 1
9018 2180 7 20 P HRCI 1
9021 2180 7 20 P HRCI 1
9024 2180 7 20 P HRCI 1
9027 2180 7 20 P HRCI 1
9030 2180 7 20 P HRCI 1
9033 2180 7 20 P HRCI 1
9036 2180 7 20 P HRCI 1
9039 2180 7 20 P HRCI 1
9042 2180 7 20 P HRCI 1
9045 2180 7 20 P HRCI 1
9048 2180 7 20 P HRCI 1
9051 2180 7 20 P HRCI 1
9054 2180 7 20 P HRCI 1
9057 2180 7 20 P HRCI 1
9060 2180 7 20 P HRCI 1
9063 2180 7 20 P HRCI 1
9066 2180 7 20 P HRCI 1
9069 2180 7 20 P HRCI 1
9072 2180 7 20 P HRCI 1
9075 2180 7 20 P HRCI 1
9078 2180 7 20 P HRCI 1
9081 2180 7 20 P HRCI 1
9084 2180 7 20 P HRCI 1
9087 2180 7 20 P HRCI 1
9090 2180 7 20 P HRCI 1
9093 2180 7 20 P HRCI 1
9096 2180 7 20 P HRCI 1
9099 2180 7 20 P HRCI 1
9102 2180 7 20 P HRCI 1
9105 2180 7 20 P HRCI 1
9108 2180 7 20 P HRCI 1
9111 2180 7 20 P HRCI 1
9114 2180 7 20 P HRCI 1
9117 2180 7 20 P HRCI 1
9120 2180 7 20 P HRCI 1
9123 2180 7 20 P HRCI 1
9126 2180 7 20 P HRCI 1
9129 2180 7 20 P HRCI 1
9132 2180 7 20 P HRCI 1
9135 2180 7 20 P HRCI 1
9138 2180 7 20 P HRCI 1
9141 2180 7 20 P HRCI 1
9144 2180 7 20 P HRCI 1
9147 2180 7 20 P HRCI 1
9150 2180 7 20 P HRCI 1
9153 2180 7 20 P HRCI 1
9156 2180 7 20 P HRCI 1
9159 2180 7 20 P HRCI 1
9162 2180 7 20 P HRCI 1
9165 2180 7 20 P HRCI 1
9168 2180 7 20 P HRCI 1
9171 2180 7 20 P HRCI 1
9174 2180 7 20 P HRCI 1
9177 2180 7 20 P HRCI 1
9180 2180 7 20 P HRCI 1
9183 2180 7 20 P HRCI 1
9186 2180 7 20 P HRCI 1
9189 2180 7 20 P HRCI 1
9192 2180 7 20 P HRCI 1
9195 2180 7 20 P HRCI 1
9198 2180 7 20 P HRCI 1
9201 2180 7 20 P HRCI 1
9204 2180 7 20 P HRCI 1
9207 2180 7 20 P HRCI 1
9210 2180 7 20 P HRCI 1
9213 2180 7 20 P HRCI 1
9216 2180 7 20 P HRCI 1
9219 2180 7 20 P HRCI 1
9222 2180 7 20 P HRCI 1
9225 2180 7 20 P HRCI 1
9228 2180 7 20 P HRCI 1
9231 2180 7 20 P HRCI 1
9234 2180 7 20 P HRCI 1
9237 2180 7 20 P HRCI 1
9240 2180 7 20 P HRCI 1
9243 2180 7 20 P HRCI 1
9246 2180 7 20 P HRCI 1
9249 2180 7 20 P HRCI 1
9252 2180 7 20 P HRCI 1
9255 2180 7 20 P HRCI 1
9258 2180 7 20 P HRCI 1
9261 2180 7 20 P HRCI 1
9264 2180 7 20 P HRCI 1
9267 2180 7 20 P HRCI 1
9270 2180 7 20 P HRCI 1
9273 2180 7 20 P HRCI 1
9276 2180 7 20 P HRCI 1
9279 2180 7 20 P HRCI 1
9282 2180 7 20 P HRCI 1
9285 2180 7 20 P HRCI 1
9288 2180 7 20 P HRCI 1
9291 2180 7 20 P HRCI 1
9294 2180 7 20 P HRCI 1
9297 2180 7 20 P HRCI 1
9300 2180 7 20 P HRCI 1
```



12F	C2B490	1124	JF	NZ,ERROR
112	00A5B8	1130	CALL	MB8B0
175	00A5B8	1140	LD	L,(IX+0)
1138	00A5B8	1140	LD	H,(IX+1)
914B	---	1140	LD	HL,I,A
913C	23	1170	INC	HL
913D	3600	1180	LD	(HL),#00
913F	CV	1190	RET	
9140	3D	1200	FLUSH:	DEC A
9141	C2B190	1210	JF	NZ,ERROR
9144	0D7E90	1220	LD	A,(IX+0)
9147	A7	1230	AND	A
9148	C2A7C0	1240	JF	NZ,MB8A7
9149	C0B9F6	1250	F_DTR:	CALL MB8B9
914E	3BFB	1260	JR	C,F_DTR
9150	C9	1270	RET	
9151	FE03	1280	FILL:	CP M03
9153	C2B490	1290	JF	NZ,ERRDR
9156	0D7E90	1300	LD	A,(IX+0)
9159	0DDF88	1310	CALL	MB8DE
915C	0D6E02	1320	LD	L,(IX+2)
915F	0D6E03	1330	LD	H,(IX+3)
9162	0D5E04	1340	LD	E,(IX+4)
9165	0D5E05	1350	LD	D,(IX+5)
9168	05	1360	PUSH	DE
9169	ED53C7B1	1370	LD	(MB1C7),DE
916D	22C7B1	1380	LD	(MB1C9),HL
9170	CDFOB8	1390	CALL	MB8F0
9173	22C8B1	1400	LD	(MB1CB),A
9176	3C	1410	INC	A
9177	CDE4B8	1420	CALL	MB8E4
917A	CD11B0	1430	CALL	MB011
917D	3E04	1440	LD	A,M04
917F	3B86	1450	JR	C,F_PAS1
9181	3E02	1460	LD	A,M02
9183	2B02	1470	JR	Z,F_PAS1
9185	3E01	1480	LD	A,M01
9187	32CCB1	1490	F_PAS1:	LD (MB1CC),A
918A	CD0C91	1500	F_PAS2:	CALL F_PAS5
918D	0D42	1510	SBC	HL,BC
918F	CD0A91	1520	CALL	S_PAS4
9192	2BF6	1530	JR	Z,F_PAS2
9194	D1	1540	PDP	DE
9195	ED53C7B1	1550	LD	(MB1C7),DE
9199	CD0C91	1560	F_PAS3:	CALL F_PAS5
919C	09	1570	ADD	HL,BC
919D	CD0A91	1580	CALL	S_PAS4
91A0	2BF7	1590	JR	Z,F_PAS3
91A2	C9	1600	RET	
91A3	22C7B1	1610	S_PAS4:	LD (MB1C7),HL
91A6	E8	1620	EX	DE,HL
91A7	2AC9B1	1630	LD	HL,(MB1C9)
91AA	CDFOB8	1640	CALL	MB8F0
91AD	21CB81	1650	LD	HL,MB1CB
91B0	BE	1660	CP	(HL)
91B1	C9	1670	RET	
91B2	ED5B17B1	1680	F_PAS6:	LD DE,(MB1C7)
91B5	E5	1690	PUSH	HL
91B7	CDFOB8	1700	CALL	MB8F0
91B8	21CB81	1710	LD	HL,MB1CB
91BD	BE	1720	CP	(HL)
91BE	E1	1730	PDP	HL
91BF	C9	1740	RET	
91C0	2AC9B1	1750	F_PAS5:	LD HL,(MB1C9)
91C3	23	1760	F_PAS7:	INC HL
91C4	23	1770	INC	HL
91C5	CD0B91	1780	CALL	F_PAS6
91C8	2BF9	1790	JR	Z,F_PAS7
91CA	2B	1800	DEC	HL
91CB	2B	1810	DEC	HL
91CC	E5	1820	PUSH	HL
91CD	2AC9B1	1830	LD	HL,(MB1C9)
91DA	2B	1840	F_PAS8:	DEC HL
91D1	2B	1850	DEC	HL
91D2	CD0B91	1860	CALL	F_PAS6
91D5	2BF9	1870	JR	Z,F_PAS8
91D7	23	1880	INC	HL
91D8	23	1890	INC	HL
91D9	ED5B17B1	1900	LD	DE,(MB1C7)



91DD	CD0B8B	1910	CALL	MB8C0
91E0	E1	1920	PDP	HL
91E1	ED5B17B1	1930	LD	DE,(MB1C7)
91E5	CDFOB8	1940	CALL	MB8F6
91E8	2AC9B1	1950	LD	HL,(MB1C7)
91EB	AF	1960	XDR	A

91EC	ED4BCCB1	1970	LD	BC,(MB1CC)
91F0	47	1980	LD	B,A
91F1	C9	1990	RET	
91F2	FE01	2000	CP	I
91F4	C2B490	2010	JP	NZ,ERROR



91F7	3E0A	2020	LD	A,M0A
91F9	1B07	2030	JR	PAS
91FB	FE01	2040	CP	I
91FD	C2B490	2050	JP	NZ,ERROR
9200	3E0B	2060	LD	A,M0B
9202	3296A0	2070	LD	(MB096),A
9205	0D6E00	2080	LD	L,(IX+0)
9208	0D6E01	2090	LD	H,(IX+1)
9208	46	2100	LD	B,(HL)
920C	23	2110	INC	HL
920D	5E	2120	LD	E,(HL)
920E	23	2130	INC	HL
920F	56	2140	LD	D,(HL)
9210	E8	2150	EX	DE,HL
9211	7E	2160	LD	A,(HL)
9212	E5	2170	PUSH	HL
9213	C5	2180	PUSH	BC
9214	CD0A9B	2190	CALL	MB8A5
9217	CD0B89	2200	CALL	MB906
921A	E8	2210	EX	DE,HL
921B	0E0B	2220	LD	C,M0B
921D	3A96A0	2230	LD	A,(MB096)
9220	F5	2240	PUSH	AF
9221	FE0A	2250	CP	M0A
9223	CD0B92	2260	CALL	Z,D,M0A
9226	CD0992	2270	CALL	NZ,J,FUTM
9229	CD0989	2280	CALL	MB9A9
922C	3EFF	2290	LD	A,MFF
922E	2197A0	2300	LD	HL,MB097
9231	CD0B8B	2310	CALL	MB8AB
9234	3EFF	2320	LD	A,MFF
9236	CD0A8B	2330	CALL	MB85A
9239	F1	2340	PDP	AF
923A	CD0A8B	2350	CALL	MB85A
923D	3E0B	2360	LD	A,M0B
923F	CD0A8B	2370	CALL	MB85A
9242	C1	2380	PDP	BC
9243	E1	2390	PDP	HL
9244	23	2400	INC	HL
9245	10CA	2410	DJNZ	BUCLE
9247	C9	2420	RET	
9248	2197A0	2430	D_RDTA:	LD HL,MB097
924B	1A	2440	LD	A,(DE)
924C	060B	2450	LD	B,M0B
924E	17	2460	R_BUC1:	RLA
924F	CB1E	2470	RR	(HL)
9251	23	2480	INC	HL
9252	10FA	2490	DJNZ	R_BUC1
9254	13	2500	INC	DE
9255	0D	2510	DEC	C
9256	2BF0	2520	JR	NZ,D_RDTA
9258	C9	2530	RET	
9259	219FA0	2540	U_RDTA:	LD HL,MB09F
925C	1A	2550	LD	A,(DE)
925D	060B	2560	LD	B,M0B
925F	2B	2570	R_BUC2:	DEC HL
9260	17	2580	RLA	
9261	CB16	2590	RL	(HL)
9263	10FA	2600	DJNZ	R_BUC2
9265	13	2610	INC	DE
9266	0D	2620	DEC	C
9267	2BF0	2630	JR	NZ,U_RDTA
9269	C9	2640	RET	
926A	FE04	2650	RIGHT:	CP 4
926C	C2B490	2660	JP	NZ,EPRDR
926F	3E01	2670	LD	A,1
9271	32E192	2680	LD	(12DER),A
9274	CD0B92	2690	CALL	SCROLL
9277	C9	2700	RET	
9278	FE04	2710	LEFT:	CP 4
927A	C2B490	2720	JP	NZ,ERRDR
927D	3E00	2730	LD	A,0
927F	32E192	2740	LD	(12DER),A
9282	CD0B92	2750	CALL	SCROLL
9285	C9	2760	RET	
9286	DD7EAA	2770	SCPDLL:	LD A,(IX+6)
9289	32E092	2780	LD	(ANCHD),A
928B	2100C9	2790	LD	HL,2048
928F	1600	2800	LD	D,0
9291	DD5E06	2810	LD	E,(IX+6)
9294	ED53	2820	HL,DE	
9296	22E492	2830	LD	(PASD),HL

9299	210AA0	2840	LD	HL,2048
929C	19	2850	ADD	HL,DE
929D	22E292	2860	LD	(PASD),HL
92A0	DE6E00	2870	LD	L,(IX+0)
92A3	DD6E02	2880	LD	H,(IX+2)
92A6	3AE192	2890	LD	A,(12DER)
92A9	FE00	2900	CF	0
92AB	2B07	2910	JR	NZ,P_PASA
92AD	DD7E04	2920	LD	A,(IX+4)
92B0	CD0B92	2930	CALL	12Q
92B3	C9	2940	RET	
92B4	DD7E04	2950	P_PASA:	LD A,(IX+4)
92B7	CDE692	2960	CALL	DER
92B8	L9	2970	RET	
92B8	F5	2980	12Q1:	PUSH AF
92BC	23	2990	INC	HL
92BD	E5	3000	PUSH	HL
92BE	CD1ABC	3010	CALL	MB01A
92C1	060B	3020	LD	B,0
92C3	C5	3030	BUC1:	PUSH BC
92C4	E5	3040	PUSH	HL
92C5	D1	3050	PDP	DE
92C6	23	3060	INC	HL
92C7	060B	3070	LD	B,0
92C9	3AE092	3080	LD	A,(ANCHD)
92CC	3D	3090	DEC	A
92CD	4F	3100	LD	C,A
92CE	1A	3110	LD	A,(DE)
92CF	ED0B	3120	LDIR	
92D1	12	3130	LD	DE,A
92D2	ED4BE492	3140	LD	BC,(PASD)
92D6	09	3150	ADD	HL,BC
92D7	C1	3160	PDP	BC
92D8	10C9	3170	DJNZ	BUC1
92DA	E1	3180	PDP	HL
92DB	F1	3190	PDP	AF
92DC	3D	3200	DEC	A
92DD	2BDC	3210	JR	NZ,12Q
92DF	C9	3220	RET	
92E0		3230	ANCHD:	DEFS 1
92E1		3240	12DER:	DEFS 1
92E2		3250	PASD1:	DEFS 2
92E4		3260	PASD0:	DEFS 2
92E6	F5	3270	DER:	PUSH AF
92E7	23	3280	INC	HL
92E8	E5	3290	PUSH	HL



92E9	CD1ABC	3300	CALL	MB01A
92EC	060B	3310	LD	B,0
92EE	C5	3320	D_BUC1:	PUSH BC
92EF	E5	3330	PUSH	HL
92F0	D1	3340	PDP	DE
92F1	2B	3350	DEC	HL
92F2	060B	3360	LD	B,0
92F4	3AE092	3370	LD	A,(ANCHD)
92F7	3D	3380	DEC	A
92F8	4F	3390	LD	C,A
92F9	1A	3400	LD	A,DE
92FA	ED0B	3410	LDIR	
92FC	12	3420	LD	DE,A
92FD	ED4BE292	3430	LD	BC,(PASD)
9301	09	3440	ADD	HL,BC
9302	C1	3450	PDP	BC
9303	10C9	3460	DJNZ	D_RDTA
9305	E1	3470	PDP	HL
9306	F1	3480	PDP	AF
9307	3D	3490	DEC	A
9308	2BDC	3500	JR	NZ,DEP
930A	C9	3510	RET	

## ETIQUETAS

ANCHD	92E0	BUC1	92C3	BUCLE	9211
BUFFE	90B0	DER	92E6	DOWN	91F2
D_BUC	92EE	D_RDTA	9246	ERROR	90B4
E_BUC	90B7	FILL	9151	FLUSH	9140
F_OTP	9148	F_PAS1	9187	F_PAS2	918A
F_PAS3	9199	F_PAS5	91C0	F_PAS1	91B2
F_PAS7	91C3	F_PAS8	91D0	GET	912E
GPAP	9107	GPEN	90F1	12DER	92E1
12Q	92B8	LEFT	9278	NAME	903B
PAS	9202	PASD	92E4	PAS01	92E2
P_PASA	92B4	RESET	9111	RIGHT	92E6
R_BUC1	924E	R_BUC2	925F	SCROLL	92B6
SETCL	90CC	S_PAS	90E2	S_PAS4	9133
TABLA	90B9	TTER	90C0	UP	91F8
U_RDTA	9259				



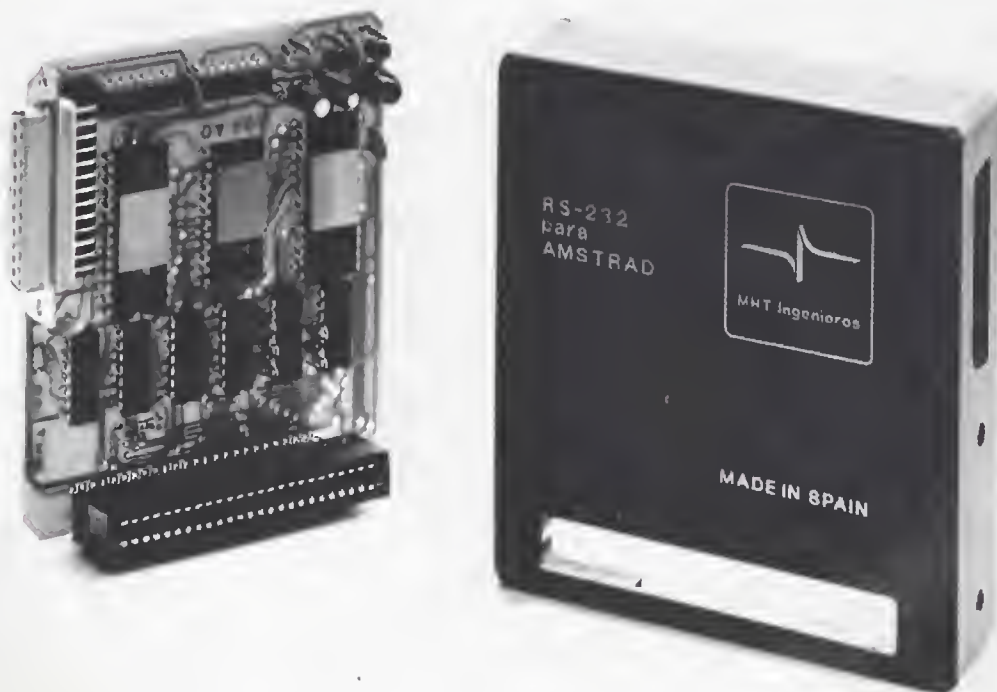
# EMPEZAMOS A POTENCIAR TU AMSTRAD

## NUEVO INTERFACE RS232

Permite comunicar los ordenadores Amstrad CPC 464, CPC 664 y CPC 6128, con impresoras y plotters con entrada serie, modems y otros ordenadores

## CARACTERISTICAS TECNICAS

- Salida serie RS-232C estándar
- Software contenido en Eprom, por lo que no ocupa memoria del usuario. Genera comandos para facilitar el uso desde el basic.



## ...TAMBIEN PERIFERICOS PARA AMSTRAD

Es un producto desarrollado y fabricado en España por:



**PRODUCTOS DISTRIBUIDOS: LSB, S.A.**

Sánchez Pacheco, 78 - 28002 MADRID - TEL.: 413 92 68



# MANTENGA SU AMSTRAD COMO NUEVO CON ESTA PRACTICA FUNDA.

POR SOLO: **2.250** ptas.

Ahora puede recibir la suya.  
Rellene el cupón y envíelo a:  
BAZAR POPULAR  
Apartado 27.500  
08080 BARCELONA



Deseo recibir el siguiente pedido:

- ☐ Funda AMSTRAD 464. 2.250 ptas.
- ☐ Funda AMSTRAD 664. 2.250 ptas.
- ☐ Funda AMSTRAD 6128. 2.250 ptas.

Indique su monitor: ☐ Verde ☐ Color

Disponemos también de los siguientes modelos: SEIKOSHA SP-800/1000 (1.200 ptas.). SPECTRUM 16/48 (430 ptas.). SPECTRUM PLUS (560 ptas.). COMMODORE 64 y VIC-20 (780 ptas.). SAGA-1 EMPEROR (650 ptas.). IMPRESORA AMSTRAD DMP-1 (1.400 ptas.). Indique la que desee.

Forma de pago: ☐ Contra-reembolso ☐ Sellos de correos adjuntos.

Gastos de envío: 150 ptas.

NOMBRE ..... EDAD .....

DOMICILIO ..... TELEF. ....

POBLACION .....

CODIGO POSTAL ..... PROVINCIA .....



ESPECIALISTAS EN SOFTWARE COMERCIAL  
Y A LA MEDIDA PARA  
Y OTROS EQUIPOS

**AMSTRAD**

## \*\* PROGRAMAS \*\*

- ADMINISTRACION DE FINCAS
- GESTION INTEGRADA  
(Facturación-Almacén-Clientes)
- ARCHIVO MULTIUSO  
(Gimnasios-Academias-Tiendas de Discos-Bibliotecas, etc.)
- ETIQUETAS

- VIDEO-CLUBS
- CONTABILIDAD-FACTURACION
- PROGRAMAS TECNICOS  
(Cálculos de estructuras: Ingeniería, Arquitectura, Andamios,  
Presupuestos y mediciones-Hormigón)
- CLINICAS (Padológicas-Veterinarias, etc.)

\*\* SEMINARIOS ESPECIALIZADOS PARA TECNICOS \*\*

\*\* CLASES INDIVIDUALES (TUTORIAS) \*\*

CONSULTE SIN COMPROMISO PRECIOS, EQUIPOS, TIEMPO DE REALIZACION, CURSOS, ETC.

**ZURBANO, 4**

**☎ 410 47 63**

**28010 MADRID**





DISTRIBUIDORES DE PRODUCTOS INFORMATICOS Y ELECTRONICOS

Ponemos a tu disposición el mejor **Ordenador** para tus necesidades, con un asesoramiento en software y hardware, para sacarle el mayor partido.

**TE ESPERAMOS**

Hermosilla, 75 - 1.º-14. 28001 MADRID. Tels. 276 43 94/435 04 70

## MASTER COMPUTER



*Si no lo encuentras en tu tienda habitual, llámanos y te lo enviaremos directamente contra reembolso.*

Tenemos todos los modelos de AMSTRAD, periféricos, software y libros. Programas y juegos para el 8256.

AMSTRAD (programas e instrucciones en castellano)

Comodore  
Apple-Accorn  
Spectrum  
Robot

**Fischertechnik**

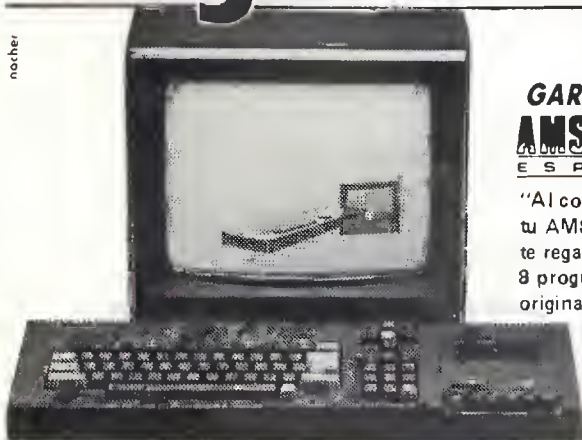
precio incluyendo caja de construcción software, interface y adaptador **34.990** ptas.

Distribuidor para España

Centro Comercial, Local 15. Ciudad Sto. Domingo  
Carretera de Burgos, Km 28 ALGETE · MADRID  
Telf. 622 12 89

**Ya tenemos el  
ATARI 520 S.T.**

## Sigue la línea del futuro



**GARANTIA  
AMSTRAD  
ESPAÑA**

"Al comprar  
tu AMSTRAD  
te regalamos  
8 programas  
originales"

**FUNDAS PARA  
AMSTRAD**

ORDENADOR AMSTRAD 464 F. Verde  
ORDENADOR AMSTRAD 6.128 F. Verde

DISTRIBUIDOR INDESCOMP PARA ORENSE:



**Almacenes Méndez**

CAPITAN CORTES, 17. TELEF.- 228607. ORENSE

Servimos a tiendas y almacenes

Presenta este anuncio y obtendrás un **OBSEQUIO** en tu compra



# MICRO-1

Duque de Sesto, 50. 28009 Madrid. Tel. 275 96 16 (Metro O'Donell o Goya)

EL IVA  
LO PAGA  
MICRO-1



Super Test: 2.300 ptas



Zorro: 2.600 ptas



Gyroscope: 2.300 ptas.



Sabrewulf: 1.850 ptas.



Million: 2.500 ptas.



Hypersport: 2.300 ptas.



Tornado L. L.: 1.950 ptas.



Dynamite Dan: 2.100 ptas.

Exploding Fist  
Jump Jet  
Alien 8  
Night Lore  
Ghostbusters  
Airway Encounter  
Airway Encounter Disco

Ptas.  
2.300  
2.495  
1.875  
1.875  
1.950  
1.750  
3.300

Raid Over Moscow  
Fighter Pilot  
Master OF T. Lamp  
Nightshade  
Hacker  
Mapgame  
Tornado Low Level Disco

Ptas.  
2.300  
1.975  
1.950  
1.950  
1.950  
2.700  
3.300

**SOFTWARE**  
por cada programa  
**GRATIS**

Este magnífico  
reloj de cuarzo



Envío contra-reembolso sin ningún gasto de envío. Teléfonos (91) 275 96 16/274 53 8  
Pidiendo a MICRO-1. Duque de Sesto, 50. 28009 Madrid.



# SINCLAIR STORE

## EL CENTRO DEL HARDWARE



SPECTRUM 48 K  
SPECTRUM PLUS  
SPECTRUM 128  
SINCLAIR QL  
COMMODORE 64  
COMMODORE 128  
COMMODORE PC 10  
COMMODORE PC 20  
AMSTRAD 472  
AMSTRAD 6128  
AMSTRAD 8256  
Y .....  
SPECTRAVIDE0  
MSX

- EN SINCLAIR STORE USTED NO PAGA EL IVA
- IMPORTANTES DESCUENTOS Y/O REGALOS
- POR LA COMPRA DE UN ORDENADOR, CURSO GRATIS DE INFORMATICA
- SOFTWARE DESCUENTOS HASTA EL 20%
- MONITORES 20% DESCUENTO.
- EN TODAS LAS IMPRESORAS 20% DE DESCUENTO
- JOYSTICK QUICK SHOT II  
INTERFACE TIPO KEMPSTON 3.800 Pts.
- JOYSTICK ANATOMICO AMARILLO  
INTERFACE TIPO KEMPSTON 3.200 Pts.

- PC COMPATIBLE IBM P.V.P. 212.000 Pts.
- ¡ULTIMA NOVEDAD EN EL MERCADO!  
ATARI 520 ST YA DISPONIBLE.  
¡VEN A PROBARLO!
- PRECIOS ESPECIALES PARA COLECTIVOS Y EMPRESAS
- DISTRIBUIDORES OFICIALES DE TODAS LAS MARCAS.  
CON AUTÉNTICO SERVICIO PROFESIONAL DE POST-VENTA
- VEN A VERNOS, NOSOTROS MANTENEMOS LAS REBAJAS,  
EN TODOS LOS ARTICULOS, HASTA EL 31 DE MARZO.
- NECESITAMOS DISTRIBUIDORES.  
SOMOS MAYORISTAS

***sinclair store***

**SOMOS PROFESIONALES**

BRAVO MURILLO, 2  
(Glorieta de Quevedo)  
Tel. 446 62 31 - 28015 MADRID  
Aparcamiento GRATUITO Magallanes, 1

DIEGO DE LEON, 25  
(Esq. Nuñez de Balboa)  
Tel. 261 88 01 - 28006 MADRID  
Aparcamiento GRATUITO Nuñez de Balboa, 114

FELIPE II, 12  
(Metro Goya)  
Tel. 431 32 33 - 28 009 MADRID  
Aparcamiento GRATUITO Felipe II